UNIVERSITY OF OSLO
Department of informatics

User Experience Design Patterns for Web 2.0 Applications

Master thesis
60 credits

Knut Andreas Ruud
knutaru@student.matnat.uio.no

May 1st 2009

Abstract

The goal of the research presented in this thesis is to identify user experience patterns based on two user studies of a web-based application for collaborative story-writing. These patterns might be useful for reusing knowledge on future projects with similar goals. Although several pattern collections already exist within HCI, these collections are primarily focused on usability-related issues. The present research attempts to address a broader class of human-technology interaction aspects, including aspects related to fun, emotion, motivation and sociability.

This research is part of the Citizen Media project, which aims to make it easier for all users regardless of experience to create and share content through social network applications. The main contribution of this work is a set of user experience patterns which are based on findings from two user evaluation studies. The first study was conducted in June 2008 with 12 adults as participants. The second study was conducted in January 2009 with 35 eight year old children as participants.

In order to determine the uniqueness of the patterns identified in this research, 3 commonly available pattern collections were selected and compared with the present set of patterns. The results show that almost half of the patterns overlap to a greater or lesser extent with already existing patterns, indicating that they are supported by findings from other contexts as well.

Acknowledgements

I would like to thank my supervisor Amela Karahasanovic for her valuable guidance and encouragement. I am also grateful to SINTEF for providing me with great facilities during the writing of this thesis. Finally, special thanks go to my family and my wonderful girlfriend Linda, for always being there and supporting me.

Oslo, May 2009
Knut Andreas Ruud

Table of contents

List of tables

List of figures

1 Introduction

Section 1.1 describes the motivation for this research. The objective, as well as research method and research context is described in sections 1.2 to 1.4. The contributions of this research are presented in section 1.5. Finally, section 1.6 provides an overview of the thesis.

1.1 Motivation

User interfaces are what enables communication between humans and computers, and are an important aspect of computing. The field of HCI (Human-Computer Interaction) is concerned with finding ways to design, support, and optimize the interface between humans and machines. Much of the work done within HCI has focused on optimizing easily quantifiable aspects of the interaction process, like effectiveness of operation and proneness to errors, in order to support and optimize day-to-day tasks encountered in the workplace. During the last decade, the use of computers at home for social and entertainment purposes has become increasingly widespread. This is demonstrated by the success of SNS's (Social Networking Sites) such as MySpace, Facebook, Cyworld and Bebo (Boyd & Ellison, 2008). In this context, users are engaged in an application not because they need to, but because they want to. Consequently, while designing such systems, there is a shift of focus from effectiveness to enjoyment. How can a system support users that want to be entertained? Such questions imply a shift from the dominantly task- and work-oriented usability paradigm traditionally adopted in HCI (Hassenzahl & Tractinsky, 2006), to a more experiential perspective commonly referred to as user experience. While a widely accepted and shared understanding of user experience is still lacking (Hassenzahl, 2008), the user experience perspective generally implies a shift of focus to user emotion, sensation, and the meaning as well as value of human-technology interactions (Law, Roto, Hassenzahl, Vermeeren, & Kort, 2009).

Designing a positive user experience is a complex task requiring knowledge from several disciplines. To this end, patterns are a method for communicating solutions to recurring design problems in a way which facilitates reuse. Patterns were initially used in architecture (Alexander, 1977), but have subsequently been adopted in several other fields such as pedagogy (Bennedsen & Eriksen, 2003), software engineering (Gamma, Helm, Johnson, & Vlissides, 1994), and HCI (Borchers, 2001; Duyne, Landay, & Hong, 2006; Tidwell, 1999, 2005; Van Welie, 2008). The design patterns currently available in HCI focus on designing user interfaces that maximize usability. However, there is presently a lack of patterns that address aspects related to user experience, which might be particularly useful in the design of social network applications.

1.2 Objective

The objective of this research was to identify UX (User Experience) patterns that support the creation of social network applications. This was done by analyzing observations from two user studies on an application for co-creation and sharing of content (Lüders, Karahasanovic, Pelt, & Herreweghe, 2008; Wurhofer et al., 2008). Typical questions that were addressed by these patterns include:

- How can users be motivated to share content online?
- How can one enable users to co-operate during content creation?
- What sort of features do users expect in social applications?

1.3 Research method

Two studies have been conducted where two different groups of users have evaluated an internet-based application for cooperative writing of stories. One research group has analyzed the results in order to provide feedback for improvement of the application itself (Lüders et al., 2008; Wurhofer et al., 2008), while I have analyzed the results in order to identify patterns based on successful solutions to issues that were encountered.

1.4 Research context

This master thesis is part of the Citizen Media project. Citizen Media is a 3-year research project launched in 2006 which involves 16 partners from across Europe.

The Citizen Media project aims to develop applications that make it possible for non-professional users to create and share content with other users in a way that resembles offline communities. Although user generated media have become increasingly widespread during the last decade, studies indicate that there exists a divide between those who consume and those who produce online media (Verde et al., 2008). The Citizen Media project aims to make this divide smaller.

1.5 Contribution

The main contributions of this research are listed below.

- Development of guidelines for writing patterns
- Analysis of Citizen Media user experience patterns
- Identification of new user experience patterns
- Comparison with existing pattern collections

1.5.1 Development of guidelines for writing patterns

I created a set of guidelines for writing patterns, based on Karahasanovic et al. (2008). These guidelines provided more detailed guidance on the form and content of patterns than what was given in the original definition.

1.5.2 Analysis of Citizen Media user experience patterns

I analyzed 26 user experience patterns from an earlier phase of the Citizen Media project (Karahasanovic et al., 2008), based on the guidelines identified previously. This analysis made visible some problems with the content of the existing patterns, from which I prepared a set of recommendations for how they could be improved. The lessons learned from this analysis informed subsequent phases of this research.

1.5.3 Identification of new user experience patterns

I performed an analysis of the findings from two user evaluation studies of the Citizen Media application Talhonia (Lüders et al., 2008; Wurhofer et al., 2008), which led to the identification of 12 user experience patterns. I then performed a revision and merging of patterns that were identified in an earlier phase of the project (Karahasanovic et al., 2008), expanding the initial set of patterns to 16.

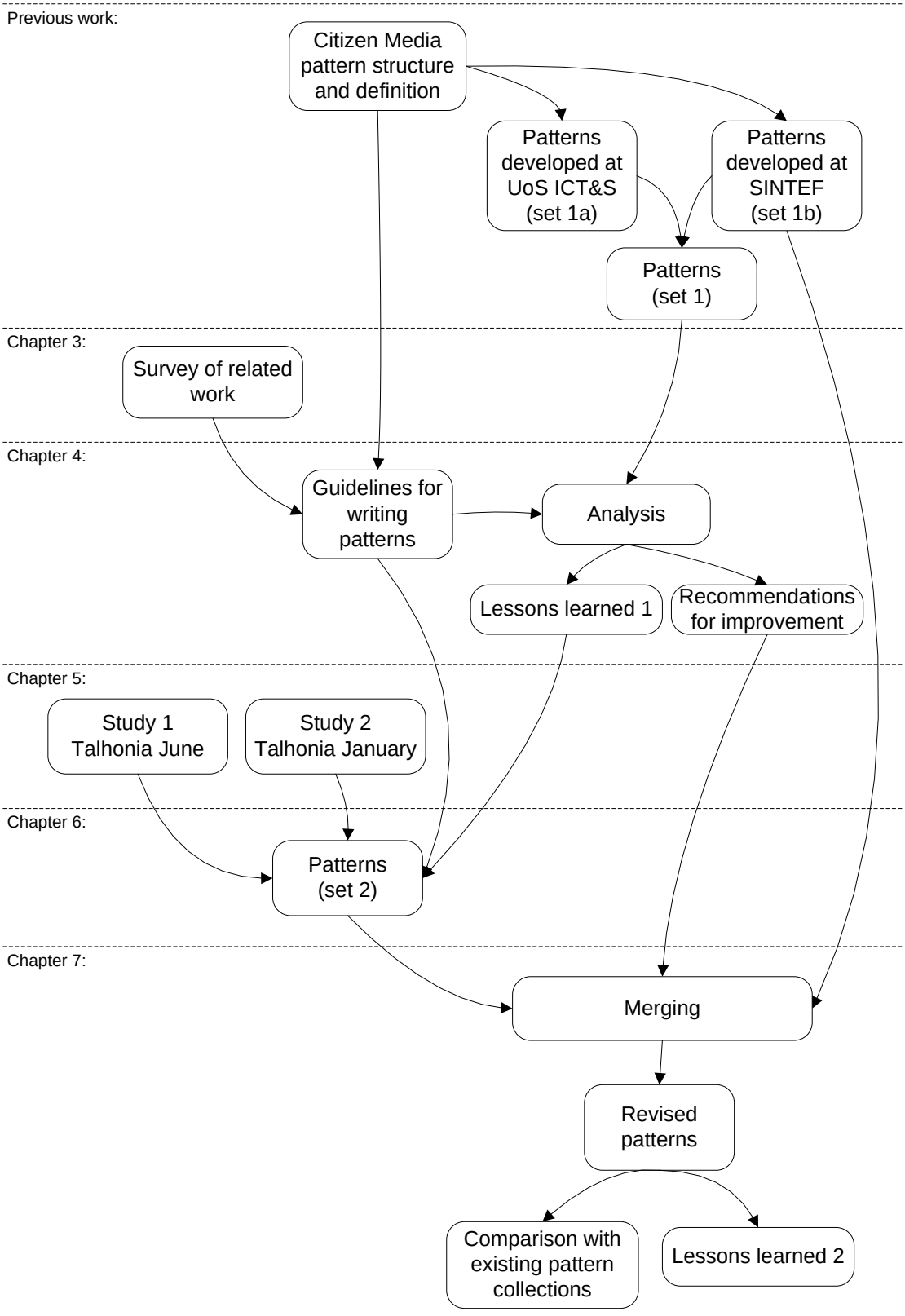
1.5.4 Comparison with existing pattern collections

To determine the uniqueness of the newly identified patterns, I conducted a comparison of the final set of 16 user experience patterns with 3 external pattern collections (Duyne et al., 2006; Tidwell, 1999; Van Welie, 2008), comprising a total of 291 patterns. The comparison indicated that there is some degree of overlap on 4 patterns, high degree of overlap on 3 patterns, and no overlap on the remaining 9 patterns. These results suggest that many of the patterns address issues that have been observed elsewhere, supporting the premise the pattern is based on.

1.6 Thesis overview

This section gives an overview of the work that is described in this thesis. Figure 1 gives a visual overview of the research process, followed by a chapter overview.

Figure 1: Research overview



The remainder of the thesis is organized as follows:

Chapter 2 Design patterns in HCI	This chapter gives a brief history of design patterns, and presents an overview of how patterns are used in HCI.
Chapter 3 Previous work on pattern identification and evaluation	This chapter presents an overview of previous work related to identification and evaluation of design patterns.
Chapter 4 Design patterns in Citizen Media	This chapter presents detailed guidelines for pattern writing, based on the pattern definition in Karahasanovic et al. (2008). These guidelines were subsequently used in the analysis of design patterns that were identified in an earlier phase of the Citizen Media project (patterns set 1, in Figure 1). Recommendations for improvement are suggested, and lessons learned from the analysis are summarized at the end of the chapter.
Chapter 5 UX patterns based on Talhonia studies: Research method	This chapter describes the two user evaluation studies that were conducted on the Citizen Media application Talhonia (Lüders et al., 2008; Wurhofer et al., 2008), and my subsequent analysis of the findings. This analysis formed the basis for identifying new user experience patterns.
Chapter 6 UX patterns based on Talhonia studies: Results	This chapter presents the new user experience patterns that were identified based on the analysis described in chapter 5 (patterns set 2, in Figure 1).
Chapter 7 Merging SINTEF UX patterns	In this chapter, the patterns identified at SINTEF previously (set 1b, in Figure 1) were merged with the new patterns presented in chapter 6 (set 2, in Figure 1). This was done in order to form the existing patterns into a coherent collection of patterns. This process resulted in a final set of 16 patterns, which are summarized and discussed in light of pattern collections that are currently in use within HCI.
Chapter 8 Conclusions and future work	This chapter concludes the thesis, summarizes the contributions of this research, and briefly discusses possibilities for further research.

Appendix A
List of CM patterns — This appendix contains the 26 patterns that were identified in an early phase of the Citizen Media project, and which are analyzed in chapter 4.

Appendix B
Study 1 findings — This appendix contains the findings of the Talhonia June study.

Appendix C
Study 2 findings — This appendix contains the findings of the Talhonia January study.

2 Design patterns in HCI

This chapter presents a brief history of design patterns, and gives a general overview of how design patterns are used in HCI. The relationship between design patterns and other forms of guidance such as guidelines and claims are discussed, as well as applications of design patterns. Following this, a more detailed definition of patterns is given through a presentation of HCI pattern characteristics and presentation formats. Concluding this chapter is a presentation of two example patterns which demonstrate the differences in two different pattern formats that are used in HCI.

2.1 What is a design pattern?

A design pattern presents a general solution to a recurring problem in a specific context. Ideally, a pattern should be sufficiently specific to enable implementation of the proposed solution, while abstracting the more mundane details of one specific implementation. Furthermore, patterns are seen as a way of organizing knowledge related to a specific domain, in a manner which supports reuse. One of the core ideas underlying the patterns philosophy is that it should provide a mechanism for organizing, as well as communicating knowledge. By relating patterns to each other in a hierarchical manner, pattern languages are formed.

2.2 A brief history of design patterns

The modern concept of design patterns was introduced by Alexander in his book "A Pattern Language" (1977)[1]. It contains a pattern language for the planning of towns and the construction of buildings. Alexander's language is hierarchical, beginning at the level of cities, then neighborhoods, houses, and finally the specific details of buildings such as doors and windows. Each pattern is linked to related patterns, some more general (at a higher level in the hierarchy), and some more specific (at a lower level of the hierarchy). Thus the entire hierarchy can be traversed in different ways while tackling problems at different scales.

The idea of design patterns has been adopted in other disciplines as well, one of them being software engineering. The first efforts to this end were done in the late eighties, and design patterns gradually gained attention in the software engineering community (Borchers, 2001). By encouraging reuse, design patterns were promised to give more flexible and efficient systems that would be cheaper to develop and maintain. More specifically, the main advantages that have been claimed for design patterns in software engineering include improving programmer productivity and software quality, helping novices in increasing their design skills, and improving communication among developers (Prechelt, Unger-Lamprecht, Philippsen, & Tichy, 2002).

Within HCI, the interest in design patterns is growing, as evidenced by the number of pattern collections made available over the last years (Borchers, 2001; Duyne et al., 2006; Tidwell, 1999, 2005; Van Welie, 2008).

[1] Although patterns have been in use for centuries prior to this (Borchers, 2001).

2.3 Related terms

Van Welie, Van Der Veer, and Eliëns (2000) compare patterns to guidelines, and find several limitations with guidelines. Because of their simplicity, guidelines are often too abstract and thus fail to provide any practical value. In other words, the advice given in guidelines can often be difficult to implement. Additionally, selecting the appropriate guidelines can be difficult because they are usually numerous and often offer contradicting advice. Furthermore they do not contain explicit information about which type of context is appropriate for their use. Well-formed patterns overcome these limitations because they contain information about what sort of context they can be applied in. When several patterns are collected into a language, they are required to explicitly state their relationship with other patterns. This makes it easier for the user of the patterns to find the correct pattern, and furthermore reduces the chance of patterns that offer contradicting advice. Since patterns are much more specific than guidelines, they should also be easier to implement. However, the quality of design patterns vary, and for this discussion a collection of well-formed patterns is assumed.

Dearden and Finlay (2006) compare patterns to claims and conclude that they are quite similar, the main difference being that patterns are grounded in multiple examples of successful use, while claims to a larger extent are grounded in theory. In addition, patterns state a specific problem and context.

2.4 Pattern applications and benefits

The potential for design patterns to capture design knowledge in a way which facilitates reuse is a basic assumption underlying the design patterns philosophy. Within HCI, other benefits that have been claimed for design patterns include simplifying communication, as well as being useful in education. This section briefly discusses these claims. For a look at whether such claims have any support in empirical evidence, see section 3.2.2.

Communication

Patterns might be appropriate as an aid for simplifying communication. Some authors have suggested the possibility of using patterns as a lingua franca (common vocabulary) within the field of HCI (Bayle et al., 1998; Erickson, 2000), and also across multiple fields (Borchers, 2000; Seffah, 2003). The latter is particularly welcome in the HCI field as decisions often need to be based on input from different groups of people, including designers, users, programmers, and managers. For instance in participatory design, where users are actively involved in shaping the user interface, patterns can potentially simplify communication between users and designers. By using patterns as a lingua franca, users and designers can focus on the core of their task, which is to discuss possible solutions to problems, rather than spending precious time explaining mundane details of the proposed solutions.

Education

Since patterns represent concrete pieces of knowledge in a distilled form, they might be useful for knowledge transfer. Indeed, the notion of learning “by example” is intuitively appealing, and several authors suggest design patterns to be a useful aid in education (Borchers, 2001; McGee, 2007; Seffah, 2003).

2.5 Characteristics of patterns

Although design patterns vary widely in form and structure, there are also several commonalities. Winn and Calder (2002) identified nine essential characteristics of patterns used within the field of software engineering. Dearden and Finlay (2006) discuss these characteristics from a HCI perspective. In the following, a total of 10 characteristics of HCI design patterns will be presented. Table 1 summarizes the characteristics that have been identified, as well as who they were identified by. Following this is a more detailed description of each characteristic.

Table 1: Pattern characteristics in HCI

No.	Characteristic	Source
1	A pattern implies an artifact	Winn and Calder, 2002; Dearden and Finlay (2006)
2	A pattern is both functional and nonfunctional	Winn and Calder, 2002; Dearden and Finlay (2006)
3	A pattern is part of a language	Winn and Calder, 2002; Dearden and Finlay (2006)
4	A pattern is validated through use	Winn and Calder, 2002; Dearden and Finlay (2006)
5	A pattern cannot be used meaningfully outside its domain	Winn and Calder, 2002; Dearden and Finlay (2006)
6	Patterns support a lingua franca	Dearden and Finlay (2006)
7	Patterns deal with problems at different scales	Dearden and Finlay (2006)
8	Patterns reflect design values	Dearden and Finlay (2006)
9	Patterns capture design practice	Dearden and Finlay (2006)
10	Patterns are framed in a positive way	Kotzé, Renaud, & Biljon (2008)

1. A pattern implies an artifact

Patterns should present a high level picture of the solution it describes. This issue relates to timelessness, specifically that the core idea (the solution) in a pattern should be presented in such a way that it is independent of a particular technology or interaction paradigm (like a keyboard and mouse, for instance). Winn and Calder suggest that a pattern that cannot be drawn does not embody a physical understanding of an artifacts structure, and therefore cannot be classified as a pattern.

2. A pattern is both functional and nonfunctional

A pattern should address not only the solution, but also the underlying rationale. Winn and Calder refer to these as respectively the functional and nonfunctional aspects of a pattern. This can be related to the first point concerning robustness and timelessness of patterns. Even when the technology for which the pattern was originally intended is completely renewed, access to the "why" aspects of a pattern makes it easier to update the solution.

3. A pattern is part of a language
By specifically linking to other related patterns, patterns become more powerful because they can be used to generate solutions to larger scale problems. This approach also aids selection of patterns. If you are unsure whether the pattern you have currently selected is the right one for the task, you can look at the patterns it references and decide if another pattern is a better match for the problem at hand.

4. A pattern is validated through use
When a pattern is put to use, it is implicitly evaluated and judged as giving an appropriate solution to the task at hand. This implies that the prevalence of any given pattern gives an indication of its usefulness.

5. A pattern cannot be used meaningfully outside of its domain
Patterns are not expected to yield good results when used in domains they weren't designed for. Neither are patterns that belong to different languages, because their relationship is unclear. In addition, most pattern languages are presented differently, due to the large number of pattern formats in use, which might further impede pattern use across different collections/languages.

6. Patterns support a lingua franca
In the field of HCI particularly, there is a need for a lingua franca which can aid communication between different types of people, including users, software developers, designers, managers, and evaluators among others.

7. Patterns deal with problems at different scales
Achieving the right granularity of patterns ensures that they can be used more flexibly. Patterns should be as cohesive as possible, and address problems at different scales so as to ensure maximum potential for reuse. In practice, this is a tricky aspect of pattern creation which might need some refinement over time to get right.

8. Patterns reflect design values
Patterns are not neutral; they reflect the design values, priorities, and motivations of the writer.

9. Patterns capture design practice
Patterns are identified from actual usage scenarios rather than purely theoretical or conceptual proposals.

10. Patterns are framed in a positive way
There are only a few isolated examples of anti-patterns[2] used within HCI. The prevailing view is that patterns should communicate good rather than bad design practices. Kotzé et al. (2008) argue that anti-patterns give way to serious pitfalls when used for teaching novices interaction design. A pattern is pedagogically effective because it suggests a specific solution to a given

[2] An anti-pattern presents a bad solution to a problem. The solution presented is often one which initially might appear attractive, but which backfires badly when applied.

problem, rather than warning against something one should not do, which may include a large number of different scenarios.

2.6 Presentation formats

Patterns are usually constructed from some generally accepted components. Virtually any pattern includes a description of the problem, the solution, the context in which the solution applies, and examples of successful use. There are however several differences in how patterns are presented. Fincher (2000) and Fowler (2006) identified categories of pattern formats employed in various disciplines. Based on this work, I have identified three main categories of patterns that are used within the HCI community.

Table 2: Categories in HCI, based on Fincher (2000) and Fowler (2006).

Format	Description	Typical components	Examples
Alexandrian	This form is rooted in the original patterns developed by Alexander et al. (1977). This is a textual form where the discussion of the various components flow naturally from one to the next.	• Name • Ranking of validity • Exemplifying picture • Context • Problem • Solution • Diagram illustrating solution • Related patterns	• HCI Pattern Language (Borchers, 2001) • WU Pattern Language (Graham, 2003)
GoF	This form was popularized by Gamma, Helm, Johnson, & Vlissides' (1994) book on design patterns for software engineering[3]. They are presented in a structured manner, employing several sections each with their own heading.	• Name • Context • Problem • Forces • Solution • Rationale • Consequences • Examples • Known uses • Related patterns	• Common Ground (Tidwell, 1999) • Interaction Design Pattern Library (Van Welie, 2008)
Minimal	This format rejects the Alexandrian focus on problems and solutions, and instead relies on a minimal set of components.	• Use when • Why • How • Examples	• Designing User Interfaces (Tidwell, 2005)

[3] This book is often referred to as the "Gang of Four" book, hence the abbreviation GoF.

2.7 Pattern examples

This section presents two example patterns, one based on the GoF form, and the other based on the Alexandrian form. These examples demonstrate some differences between the various pattern formats in use. The Alexandrian format, as seen in 2.7.2, is a textual form where the different sections of each pattern are implied rather than spelled out explicitly. It starts by defining the context for the pattern, followed by the authors subjective rating of the patterns validity (indicated by one or more diamond symbols). The first section in bold describes the forces the pattern will resolve. The paragraph following this discusses the problem the pattern is addressing. After this discussion, the solution is presented, followed by the word "therefore". Finally, a diagram describing the solution visually and references to related patterns are presented. The Shopping Cart pattern, as seen in 2.7.1, includes mostly the same information, but it is divided into discrete sections with headings. This might make the pattern easier to read quickly, as it's quicker to scan the different sections when they are explicitly titled. Most of the pattern collections available in HCI adhere to this more structured presentation, indicating that such a form might be more appropriate within HCI.

2.7.1 Shopping Cart

Shown below is the Shopping Cart pattern from the Interaction Design Pattern Library (Van Welie, 2008). As identified in the previous section, the format employed is loosely based on the GoF form. The sections used are *problem*, *solution* (accompanied by example), *use when*, *how*, *why*, *more examples*, and *literature*. For the sake of brevity, the two last sections which only contain references are omitted from this example.

Shopping Cart

Problem
Users want to buy a product.

Solution
Introduce a shopping cart where users can put their products in before they actually purchase them.

[image from waterpikstore.com omitted]

Use when
A site where users can browse through products and buy them. Users are not very frequent buyers and are possibly novices. For returning customers, consider a ONE-CLICK SHOPPING system. Users may buy more than one product. Users may want to select products now but pay later. Users may decide to purchase somewhere else at any time.

How
When users view a product description, they can choose to add it to their shopping cart. After adding an item to their cart, the users are shown the current contents of the cart. Users can inspect their cart contents at any time using a link that is available on every page. A persistent mini-cart could also be shown directly on the content pages. Basically the cart is a Collector that is used to collect products.

The description of the cart contents typically includes the name of the items, the quantity, availability and prices. Users can remove items from their cart if they wish and change quantities. The description of the goods is a link to the product details. Users always see the total costs of a purchase, so including [sic] shipping costs if applicable. The users must also be informed of the payment options such as which credit cards are accepted. From the cart page, the users can continue shopping or proceed with the checkout procedure. The items stay in the cart for a certain period of time, e.g. 90 days.

[example image omitted]

Why
The shopping cart is a very well known and international metaphor. This pattern allows users to gather all products first and pay for them all at once and whenever they want. By showing the total costs including shipping the users know exactly what they will have to pay when they decide to purchase. The checkout procedure using a Wizard helps users to accomplish the actual purchase with all possible assistance.

2.7.2 Closed Loop

Contrasting the above example with a more traditional Alexandrian pattern, the pattern Closed Loop from Borchers' (2001) HCI Pattern Language is shown in abridged form below.

Closed Loop

[image from the WorldBeat interactive exhibit system omitted] …you are designing an interactive exhibit or similar public system, and a very general structure of interacting with your system is in place […]. Now you need to find a way to wrap possibly many different features of your system into understandable units for the user.

◊◊◊

A public interactive system may have many features to explore and messages to convey. However, casual users will not engage with a system for a long time if they do not feel they are getting something out of it.

The WorldBeat exhibit includes features to find tunes by humming, improvise to a band, play virtual instruments, guess instrument sounds, and others, many of them with several subsections to explore and try out. It would take about an hour to work through all of its functions; this is far more time than the average user wants (and is expected) to spend with the system. To make sure that every visitor still experiences a gratifying interaction, those features are offered as alternative choices from a central selection page, exhibit structure. Upon entering one of those features, its message is explained very briefly, and after trying it out, the user is lead back to the central selection page, which he usually already knows. At this point, a user can easily leave the system, or pass on to the next waiting visitor, with a feeling of closedness because he knows that he has at least explored that feature now. [Some further discussion omitted.]

Therefore:

After two to four minutes of interaction, explain to the user what she has just seen or learned from your system, guiding her back to a central starting point in it that she recognizes and offering the opportunity to leave the system, or to continue exploring another aspect of it.

[Illustration and references to related patterns omitted.]

3 Previous work on pattern identification and evaluation

This chapter details my survey of previous work that is relevant to the research goal of this thesis. I have grouped the related work into two categories, one for work related to identification of design patterns, the other for studies related to evaluation of design patterns in HCI. Section 3.1 describes the method, and the results are described in section 3.2. Section 3.2.1 describes the results related to pattern identification, while section 3.2.2 describes the results related to pattern evaluation. The findings are summarized in section 3.3.

3.1 Method

I performed searches using several reference databases, including ISI Web of Knowledge, ACM Digital Library, IEEE Xplore, INSPEC (Ovid), CiteSeer, Scopus, Google Scholar, and CompleteSearch DBLP. The searches were performed in September 2008 and repeated in April 2009.

The following keywords were used:

- design
- pattern*
- HCI
- user experience
- guideline*
- claim*
- evaluat*
- pattern language
- interact*
- usability
- web

The search phrase "(pattern* AND (design OR language)) AND (hci OR "user experience" OR interact*)" returned 51 results in ISI Web of Knowledge. Similar searches were performed in other databases, using a combination of different keywords from the list above. These searches returned on average between 10 and 100 results, depending on the specificity of the search phrase. Irrelevant articles were filtered out by first reading the titles and then the abstracts. Additionally, several articles were identified by investigating references in the first batch of papers. I ended up with 11 articles that were closely related to my two research goals, including some papers that helped to clarify the concept and history of patterns in more general terms. The results are presented in the following section.

3.2 Results

This section describes previous work that has been done on identification of design patterns (section 3.2.1), and evaluation of design patterns in use (section 3.2.2).

3.2.1 Identification of patterns

Identifying and writing new patterns that are useful for others is a difficult task. Many pattern authors have shared their knowledge of this process in different ways, both by means of general advice and by formal definitions. Table 3 summarizes the articles I have found relating to these issues, highlighting their purpose with the article and their specific contributions to my own work.

Table 3: Work on identification of patterns

Work	Domain	Main contribution	Specific contributions relevant for my work
Borchers, 2001	HCI	Creating patterns for interactive music-based installations.	Formal definition of patterns and experiences from pattern creation.
Fowler, 2006	Design patterns	Documenting own practices for pattern creation.	Advice on many aspects of pattern writing.
Mahemoff & Johnston, 1998	HCI	Developing a pattern language based on widely used UI style guides.	Provides an underlying philosophy of usability underlying the creation of patterns.
McGee, 2007	Game design	Pattern creation and applicability to an educational setting.	Pattern creation template.
Mezaros & Doble, 1996	Software engineering	Identifying patterns for pattern creation.	Pattern language for pattern writing.
Van Welie et al., 2003	HCI	Structuring patterns in a meaningful way.	Structuring principles for pattern languages.
Segerståhl & Jokela, 2006	HCI	Evaluating usability of usability patterns.	Gives specific recommendations on making patterns easier to use.

Borchers (2001) proposed a formal model of pattern languages, describing the structure of a pattern and how several patterns can be linked to form a pattern language. In Borchers' model, each pattern is represented as a node. The references of a pattern are given by the set of edges pointing away from it, and the context of the pattern is given by the set of edges pointing towards it. Furthermore, each node (pattern) contains a set of elements defined as name *n*, ranking *r*, illustration *i*, problem *p* with forces f_1 to f_i, examples e_1 to e_j, solution *s*, and diagram *d*. Borchers' model represents a framework supporting the creation of well-formed patterns within the Alexandrian tradition. The finished pattern is intended to be presented as a continuous piece of text (in line with the Alexandrian format), but the formal representation can be useful for presentation in a hypertext environment.

Fowler (2006) documents his own experiences writing design patterns. Although written for the domain of software engineering, the advice offered might nonetheless be useful for other types of patterns.

Mahemoff and Johnston (1998) stress the importance of developing a philosophy of usability to support the creation of patterns. They develop such a philosophy based on 6 principles that must be taken into account when developing interaction patterns: efficiency (of operation), consistency (in how the interface reacts and behaves), feedback (always show the status of the system), robustness (mistakes should be easy to recover from), flexibility (tailoring the interaction according to the context), and comprehensibility (make it understandable for as many types of users as possible). Basing a collection of patterns on a predetermined set of common usability principles should help the pattern developer to create patterns that are consistent and grounded in relevant theoretical principles.

McGee (2007) used patterns in teaching game design principles to his students. To enable students to develop their own patterns, a template was created which is essentially a recipe for how to make well-structured patterns. It imposes several constraints that, when conformed to, results in clearly formulated patterns. For instance, McGee's template for writing the forces-section looks like this:

- Force 1: if a game does not have/allow [A], then players will experience problem [X].
- The word "But,"
- Force 2: if a game does have/allow [A], then players will experience problem [Y].

The idea of forces can be difficult to understand for newcomers to patterns. McGee's template serves as both an example and a description of what it should be, which makes it easy to understand the purpose of this section of a pattern.

Mezaros and Doble (1996) wrote a pattern language for pattern writing. Although much of the advice given here can also be found elsewhere, Mezaros and Doble have structured it into a pattern language that makes the information easily accessible from one resource. They provide guidance on several issues relevant for pattern writing, such as naming and organization of patterns. Although these patterns aim to support the creation of design patterns for software engineering, there are several insights that are relevant for HCI patterns as well.

Van Welie et al. (2003) identified three fundamental organizing principles for pattern languages, inspired by concepts from object oriented modeling. These are *aggregation*, *specialization* and *association*. *Aggregation* refers to how a pattern can contain other patterns that have been specialized in some way. As an example, a Shopping Cart pattern will include both a List Builder pattern for building a list of items and a Wizard pattern to describe the checkout process. *Specialization* refers to the idea that a pattern can inherit and extend properties from other patterns. Finally, *association* refers to how there can exist several alternative patterns to use in a given context. Welie et al. also identify four layers of patterns in pattern languages. Moving from top to bottom, these are *posture patterns* which describe the type or genre of page (i.e. e-commerce site), *experience patterns* which describe main user goals and tasks, *task patterns* which describe specific tasks that are part of reaching a user goal, and finally *action patterns* which is at the lowest level of the hierarchy, and describe specific objects in the interface (i.e. pushbuttons).

Segerståhl and Jokela (2006) investigated the usability of design pattern collections. They conducted a qualitative case study which aimed to explore the usability of two popular pattern collections (Tidwell's "Common Ground" collection, and Van Welie's "Interaction Design Pattern Library" collection) in the context of an industrial development project. The background for this decision was that none of the pattern collections were deemed sufficiently complete to cover all needs on the project. Using patterns from two separate collections proved to be problematic however, and based on the experiences from the project, Segerståhl and Jokela proposed 4 improvements on how collections of design patterns can be made easier to use. They are summarized below.

1. Group pattern collections based on the problem domain (activity domain), such as navigation and search.
2. Use different types of examples, including graphics, so that the reader quickly understands the core message of the pattern.
3. Name patterns so that they communicate their purpose, and base them on standardized naming conventions.
4. Integrate already existing pattern collections and compile them into a unified format, supporting easier use and making it easier to find relevant patterns.

Segerståhl and Jokela's observations suggest that more work needs to be done on standardizing patterns. Although some attempts have been made towards the creation of a standardized pattern form (Borchers, 2001; Fincher, 2003), these have been unsuccessful judging by the diversity of pattern formats currently in use.

3.2.2 Evaluation of patterns

Several authors have suggested benefits of using patterns, and the belief in design patterns as a useful design tool is reflected in the number of pattern collections already available (Borchers, 2001; Duyne et al., 2006; Tidwell, 1999, 2005; Van Welie, 2008). Of the benefits that have been claimed, the potential for design patterns to function as a lingua franca might be particularly useful in a cross-disciplinary discipline such as HCI. Furthermore, it has been suggested that design patterns can be a powerful educational aid. However, despite considerable research activity related to HCI design patterns, one important area that has received little attention is the evaluation of design patterns' usefulness in practice (Dearden & Finlay, 2006). Table 4 summarizes the research that has been conducted on evaluating the usefulness of patterns, highlighting the purpose of the studies and the method employed. Although the research has been done in various contexts and with different research goals, they are all relevant to the usefulness of patterns in interaction design.

Table 4: Studies investigating usefulness of patterns

Study	Domain	Purpose	Method
Borchers, 2001	HCI	Formal definition and creation of patterns + evaluation	Survey
Cowley & Wesson, 2005	HCI	Evaluate usefulness of patterns for interaction design	Survey
Dearden et al.,	HCI	Evaluating patterns for	Case study

2002		participatory design	
Kotzé et al. 2008	HCI	Empirically evaluating usefulness of patterns vs. anti-patterns in education	Case study and experiment

Borchers (2001) tested didactic usefulness of patterns by administering a survey to his students in a HCI design course, two weeks after a two-hour lecture on patterns. During this time students had also made use of patterns themselves in a user interface prototyping assignment. 26 out of 32 students responded, rating items on a scale from 1 (highest score) to 5 (lowest score). Overall, the students rated the usefulness of pattern languages for understanding HCI design issues as 1.96. Furthermore, they recalled on average about 1.73 patterns. The responses to these and related questions suggest that students found patterns to be useful and easy to get to grips with. However, a limitation of this study is that no comparison was made with students who had been presented with other types of design advice.

Borchers (2001) conducted an evaluation of an interactive music exhibit called WorldBeat, which was designed through active use and creation of patterns. A survey was conducted among visitors to the AEC museum[4] where the WorldBeat system was in use. Results show that the 104 participants on average rated the exhibit with the second highest score possible. These results simply show that a system based on patterns for interaction design was well-liked by users, and cannot be used to support any conclusions about the usefulness of patterns.

Borchers (2001) also found patterns to be useful for reuse in a later project. Particularly, the use of design patterns helped speed up the development process, and also aided communication. Borchers recount several occasions during the project where the use of patterns helped communication with both the client and the software developers working on the system. Again, these are purely subjective experiences, and since no comparison was made with alternative methods they cannot support any conclusions about the usefulness of patterns in general.

Cowley and Wesson (2005) conducted an experimental study on the usefulness of patterns. An experimental group used a selection of patterns to evaluate and redesign an existing website, and to design a new website. A control group performed the same task with the use of guidelines similar to the selected patterns. Although preliminary, Cowley and Wesson's results suggest that designers are more positive towards patterns than guidelines in several respects, including potential for evaluation, redesign and new design.

Dearden, Finlay, Allgar, and McManus (2002) conducted an evaluation of patterns as tools for participatory design. They invited six users to create a paper prototype of an airline and rail-travel website. The sessions lasted between 1 and 2 hours, and the participants were interviewed afterwards. Dearden et al. concluded that patterns can have a benefit in

[4] AEC: Ars Electronica Center, a technology-oriented "experience" museum.

empowering users to participate in the design process, though they do not, as Cowley and Wesson, compare the pattern approach to other alternatives.

Kotzé, Renaud, & Biljon (2008) conducted two studies to identify differences in using patterns and anti-patterns in education. The first was a case study employing a within subjects design[5] consisting of two treatments: being taught with positive guidelines, and being taught with negative guidelines. The subjects were then tested on a task. Although Kotzé et al. wanted to investigate the differences between patterns and anti-patterns, they nonetheless chose to use positive and negative guidelines in this study. The reasoning behind this arrangement was that guidelines were found to be easier to test than patterns, and furthermore that any differences between negative and positive guidelines would hold equally well for patterns and anti-patterns. Kotzé et al. admits that the study suffers from several deficiencies, including a large potential for carry-over effects due to the within-subjects design employed, but nonetheless conclude the pattern-approach to be more promising for education than anti-patterns. They conducted a more methodologically rigorous follow-up study to test their preliminary findings further, this time employing a between subjects design[6]. Results from this study show that the group of students taught with positive guidelines received a rating that was 15% higher on average than the students taught with negative guidelines. These findings indicate that positively framed guidance might be easier to learn from than negatively framed guidance.

3.3 Summary

This section summarizes the previous work done on identification of patterns (section 3.3.1), and evaluation of patterns (section 3.3.2).

3.3.1 Identification of patterns

This work concerned organization, as well as writing of design patterns.

Regarding organization of patterns, Borchers (2001) proposed a formal model of pattern languages to formalize the structure and relationships between patterns. Van Welie et al. (2003) identified three fundamental organizing principles for pattern languages, inspired by concepts from object oriented modeling. Finally, Segerståhl and Jokela (2006) investigated the use of patterns from across pattern collections and found that a common organizational principle was needed. To this end, they propose organizing patterns according to activity domain.

Related to writing patterns in more general terms, Fowler (2006) documented his own experiences writing design patterns, and formulated them as general pattern-writing advice. In a similar vein, Mezaros and Doble (1996) wrote patterns for pattern writing and collected them into a pattern language. McGee (2007) used patterns in teaching game design principles to his students. To this end, a recipe for creating well-structured patterns was prepared.

[5] In a within subjects design, each subject receives a sequence of some or all treatments, so that each subject is essentially matched with him- or herself.
[6] In a between subjects design, the subject pool is split and randomly assigned into two or more groups, and each group is then subjected to just one of the treatments.

Finally, Mahemoff and Johnston (1998) developed a philosophy of usability based on 6 principles that must be taken into account when developing interaction patterns, which are thought to help the pattern writer in creating patterns that are consistent and grounded in relevant theoretical principles. Although these articles have been written in domains ranging from game design to software engineering, they nonetheless capture some general issues that are relevant when writing design patterns within any domain.

3.3.2 Evaluation of patterns

These studies attempted to evaluate the usefulness of patterns in various contexts.

The findings from Borchers (2001) are difficult to conclude from, because they do not compare the use of design patterns with alternative solutions. It is thus difficult to determine whether design patterns lead to results that are different when compared to alternative approaches. In a similar vein, Dearden et al. (2002) attempted to investigate the usefulness of employing design patterns in a participatory design task, but did not conduct a comparison with alternative methods. Cowley and Wesson's (2005) study was the only one where the use of design patterns was compared to alternatives (guidelines). Their findings, although preliminary, suggest that the participants were more positive towards design patterns than guidelines. Kotzé et al. (2008) investigated whether patterns are more effective in education than anti-patterns. However, they chose to use guidelines rather than design patterns for the actual experiment which is puzzling, particularly in the light of Cowley and Wesson's findings, which suggest that there are differences in how design patterns and guidelines are judged by participants. Kotzé et al. cite no evidence suggesting that findings related to guidelines are directly transferable to design patterns. Therefore, their findings could perhaps be seen as more general evidence that negatively framed guidance is harder to learn from than positively framed guidance. Nonetheless, it is clear that more research is needed before any conclusions can be drawn regarding the usefulness of design patterns compared to the alternatives.

4 Design patterns in Citizen Media

Within the Citizen Media project, there already exists a set of design patterns (26 in total) based on data from earlier studies. These patterns are described in Karahasanovic et al. (2008), and are presented in Appendix A. This chapter presents an analysis of the existing design patterns in Citizen Media. The guidelines used for the analysis are given in section 4.1, and the results are presented in section 4.2. Finally, section 4.3 summarizes the lessons learned from this process.

4.1 Pattern guidelines

Analyzing the patterns found in Karahasanovic et al. (2008) was a natural first step prior to identifying new ones, in order to gain an understanding of potential problem areas. For this purpose, the pattern structure definition presented in Karahasanovic et al. was used as a starting point. This definition was then extended somewhat, in order to provide a more detailed basis for analysis. The guidelines are presented in Table 5.

For the analysis I started with the overall structure, identifying sets of patterns which presented similar ideas, and determined their degree of overlap. I then looked at the patterns one by one, assessing whether each section of the pattern conformed to the guidelines. This process resulted in a list of problem areas related to each section of the patterns, which became the basis for identifying recommendations for improvement.

Table 5: Guidelines for writing patterns

Section	Guideline
Name	Pattern names should be easy to remember in order to facilitate their use. To this end, pattern names should be kept as short as possible, while at the same time describing the gist of the pattern. Borchers (2001) suggests that a pattern name should preferably be 2 words or less, with the maximum being 4 words. In addition, consistent naming across patterns makes it easier to remember a collection of patterns. Furthermore, patterns should be easy to refer to by name in conversation[7] (Meszaros & Doble, 1996). Meszaros and Doble's pattern collection for pattern writing (1996) contains the pattern Noun Phrase Name which states that patterns should be named after the result it creates, which allows the name to be used easily in conversation[8]. Most large HCI pattern collections adhere to the Noun Phrase Name pattern (Borchers, 2001; Duyne et al., 2006; Tidwell, 1999, 2005; Van Welie, 2008). Finally, the user experience factors that are addressed by the pattern should be included along with the name.
Problem	The problem should state the basic problem that the pattern is addressing, and it should be framed as a question. Since these patterns are focused on user experience, they should primarily be framed from a user-perspective.

[7] An important feature of patterns is to foster communication by providing a Lingua Franca, therefore they should be easy to communicate both in spoken and written form.

[8] Although, initially developed to aid creation of software engineering patterns, many of Meszaros and Doble's patterns are useful for writing patterns in general.

Forces	The forces describe aspects of the design context that need to be optimized or balanced (Borchers, 2001). They can describe various trade-offs, constraints, or concerns related to the use of the pattern. Forces often come in pairs that identify what can happen if the solution is taken too far in one direction, i.e. not balanced properly (McGee, 2007).
Context	The context section of a pattern should describe when it is appropriate to apply a particular pattern (McGee, 2007). All patterns are designed to be applied within a particular context determined by one or more of the following aspects: the task itself, the user, or the environment. The context section is crucial, because for a system designer, it is difficult to find and apply a pattern without gaining a clear understanding of which types of settings it applies to.
Solution	The solution should be described specifically so that it can be implemented without raising further questions. If a solution can not be described adequately in a single pattern, it should reference other sub-patterns that describe these issues separately. Another point relating to specificity is that patterns should be testable; it should be possible to falsify a pattern if the solution does not work, so that better solutions can be developed (McGee, 2007). Borchers (2000) stress that patterns should adhere to the principles of good text design, thus one should not present the pattern reader with a set of keywords as the solution to a problem.
Examples	The examples should further support the solution by referring to commonly known implementations of the pattern. Whether an example is commonly known or not can be difficult to determine, but this must be seen in relation to the intended target audience for the patterns as well.

4.2 Findings from analysis of UX patterns

The results of my analysis of the existing Citizen Media UX patterns are detailed below. Table 6 presents an overview of the patterns that were analyzed[9]. Following this I discuss problems related to different aspects of the patterns.

Table 6: Overview of patterns for analysis

No.	UX Pattern name:
1	Forming groups of interest
2	Share Content
3	Create Content
4	Co-create Content
5	Challenge Users
6	Make Information Management Easy
7	Make It Enjoyable
8	Finding Information
9	Getting to know the application

[9] The pattern names are listed exactly as in the original document (Karahasanovic et al., 2008), thus the inconsistent use of capital letters is intentionally left as it is.

10	Provide feedback
11	Provide Personal Information
12	Successful navigation
13	Stimulate social interaction
14	Share an Experience
15	Feeling as part of the community
16	Catch the user
17	Keep the user active
18	Encourage user generated content
19	Vote to promote
20	Give information about the platform
21	Getting in touch with people sharing the same interests
22	Motivating people to make their own music available to others
23	Providing comprehensive information about a networked A/V application on its front page
24	Navigation within a networked A/V application
25	Personalization of user profiles
26	Keeping users updated

4.2.1 Overlap

Several patterns appear to address the same issues, or have solutions that overlap to a great extent. This could be remedied by either merging several patterns into one, or by making them more specific, so that the degree of overlap is reduced.

For instance, patterns 3 (Create Content) and 4 (Co-create Content) appear very similar, and even though they both address the creation of content, co-creation of content is very different from individual creation of content. More specificity is required in this case, and these two topics could each warrant a collection of patterns in order to be addressed in sufficient detail.

4.2.2 Naming

The patterns do not conform to the Noun Phrase Name pattern, opting instead to use verbs in the names. According to the Noun Phrase Name pattern, names containing verbs or prepositions are difficult to use in conversation, and should thus be avoided. In addition, using verbs in pattern names tends to make them longer. Some of the names are too long, such as pattern 15 (Feeling as part of the community) and pattern 23 (Providing comprehensive information about a networked A/V application on its front page). Furthermore, the use of verbs in a pattern name implies that a subject is present, which can be confusing. For instance, consider pattern 14 (Share an experience) and pattern 10 (Provide feedback). Pattern 14 implies that the subject is the user, since only a user of the system can share an experience. Pattern 10 on the other hand, implies that the subject is the system itself, since only the system can be meant to provide feedback. This difference in viewpoint can be seen in many of these patterns, such as patterns 1 and 3 which are named from a user perspective (Forming groups of interest; Create content), and patterns 5 and 16 which are named from a system perspective (Challenge users, Catch the user). These issues could be fixed by applying the Noun Phrase Name pattern, as well as making sure that pattern names are kept shorter than four words.

4.2.3 Problem

The problem descriptions generally conform to the requirements. There are however some exceptions and these can be found in patterns 15, 19, and 20. In these patterns, the problem statement seems to give a description of the context rather than the problem the pattern is addressing. For instance, the problem statement in pattern 20 (Give information about the platform) is given as: "Users want to be informed about the whole purpose of the platform, especially when they are novel users". This problem can be remedied by rephrasing the problem as a question. For instance, one could change it to: "How can one inform users about the purpose of the platform?"

4.2.4 Forces

Some of the patterns contain a large number of forces. As an example, pattern 1 (Forming groups of interest) contains 6 forces, and most of these are a bit too specific. The first force in pattern 1 is ok since it seems to capture a general user need (i.e. being able to meet other users with similar interests). The forces following the first one are very specific however, and are unlikely to be crucial factors for most users (e.g. forming "special interest groups" and "subscribing to special interest groups"). These more specific statements describe aspects of the solution and should be kept distinct from the forces section.

4.2.5 Context

A problem with most of the patterns is that the context section does not really communicate when the pattern should be applied. Rather, they seem to be focused on explaining why the pattern should be used, which is something that should be communicated in the forces section. For instance, the context of pattern 1 is given as "Forming of interest groups is important for networked applications in order to boost co-experience and sociability. Online interest groups reflect participation in offline communities." Reading this, one does not really learn when it is appropriate to apply the pattern. This problem can be remedied by rephrasing the context with a focus on when it is appropriate to apply the pattern in question. For example, the context in pattern 1 could be rephrased as follows: "The user wants to get in touch with other people using the service."

4.2.6 Solution

Many of the solutions presented lack specificity, and thus fail to give an adequate answer to the problem at hand. As an example, consider pattern 7 (Make it enjoyable). The solution offered here is to add challenging and surprising elements to your site. One is further advised to add enjoyable and entertaining features to the site, such as allowing users to send gifts to others. It should be evident that this solution needs more concrete details in order to be helpful for others. Furthermore, it is not possible to falsify this pattern because the solution offered is too broad in scope, which seems to be a problem with many of these patterns. How to make a site enjoyable is perhaps something which can be addressed properly with a collection of several patterns. It seems that many of the patterns share this problem, i.e. they try to offer solutions to complex, multi-faceted issues in one single pattern.

Not all of the patterns adhere to principles of good text design. For instance, the solution to pattern 16 (Catch the user) presents a bulleted list of five sentence fragments which hint at

what should be done in order to motivate the user. “Attractive design” and “promoting the platform” is suggested, which could each be the topic of an entire pattern collection in itself. “Define a clear aim/goal/purpose of the networked system” is more specific, but overlaps to a great extent with pattern 20 (Give information about the platform). Rather than providing a solution, this pattern simply lists different aspects that are recognized as important in order to motivate the user (i.e. attractive design, promotion, having a clear goal, and so on) rather than giving any kind of comprehensive guidance as to how increased user motivation can actually be achieved. Again, this seems to be due to the fact that the pattern addresses a very broad and complex issue which is impossible to account for in a single pattern.

4.2.7 Examples

Most of the examples given seem ok, but there are some exceptions. For instance, pattern 20 (Give information about the platform) gives a “negative-example” instead of a regular example. This seems a bit odd when the examples section is defined as presenting examples of successful implementation of the pattern in a system. If one cannot find a relevant successful example, it might be better to state that no examples were found.

4.3 Lessons learned

I have identified the following difficulties in writing patterns. It’s important that extra attention is paid to these issues when writing new patterns.

- Naming patterns so that they are concise and meaningful at the same time proved to be difficult. It might be necessary to do this process in several steps, starting out with a name that captures what the pattern is roughly about, and trying to find alternate ways of specifying this more concisely.
- Separating the context and forces from specific implementation details proved challenging, possibly due to confusion about the actual purpose of these parts of the pattern. A couple of basic rules of thumb, such as the ones offered in McGee (2007) could be helpful in this regard.
- Making the solution sufficiently specific as to not raise new questions appeared difficult. This problem stems at least partially from a problem with granularity. Making more patterns with less scope should make it easier to offer adequate solutions.

5 UX patterns based on Talhonia studies: Research method

This chapter describes two studies that were conducted in order to evaluate Talhonia. Section 5.1 presents an overview of the application. To give some context to the present studies, section 5.2 summarizes early work on requirements gathering and prototyping of Talhonia. The two studies are then described in section 5.3 and 5.4, respectively. The subsequent analysis of these studies led to the identification of new user experience patterns, which are described in the next chapter.

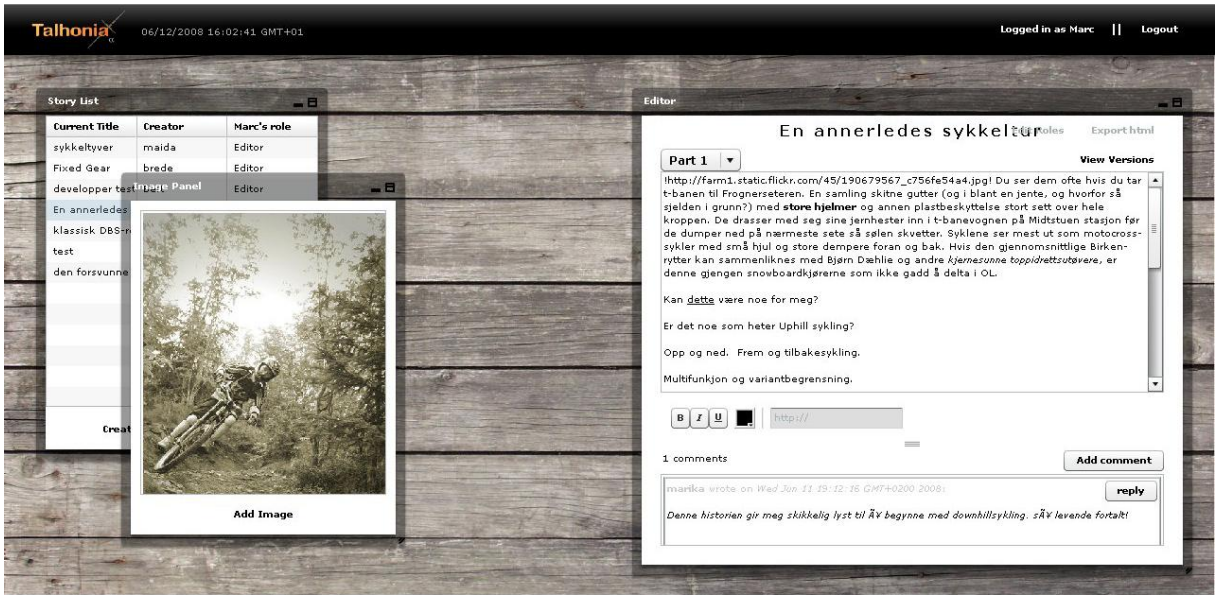
5.1 Talhonia

Talhonia is an application for collaborative storytelling, aimed at letting non-professional users share and create stories together online (Verde et al., 2008). Unlike Google Docs, which is an online word processor with asynchronous collaboration features, Talhonia is envisioned as supporting a playful shared writing-experience, where emphasis is on synchronous collaboration.

The studies described in this chapter were based on two completely different versions of Talhonia, as the application underwent a complete redesign due to much negative feedback in the first study (Lüders et al., 2008). Therefore the two versions of Talhonia, referred to as Talhonia June and Talhonia January, are presented below.

5.1.1 Talhonia June

Figure 2: Talhonia June – Viewing a story



The Talhonia June version physically resembles a table where all aspects pertaining to a story are visible at the same time. The information is split across several floating windows, which can be moved around freely by the user. The windows might also overlap each other. Only asynchronous co-writing is supported in this version, meaning that two users cannot edit a story at the same time.

5.1.2 Talhonia January

Figure 3: Talhonia January – Viewing a story

The January version of Talhonia looks more like a word processor than the previous version. It employs a fixed layout, as opposed to the user-configurable floating windows employed in the June version. The use of graphics has been toned down, and the functionality for adding images has been removed. Physical differences aside, probably the most important difference in this version is that synchronous co-writing is now supported.

5.2 Initial requirements and prototyping

The initial requirements for the application were derived from the Citizen Media project which it is part of. These relate to aspects of getting users involved in and enjoying a social application. For instance, the motivation to join an application like this depends on factors like critical mass – whether enough content is available and/or enough members are present to justify joining, integration towards existing services, and the user experience in general.

In the early phase of developing this application, three studies were conducted in order to investigate needs of non-professional users related to collaborative aspects of co-creation and sharing of content (Verde et al., 2008). The methods employed included questionnaire (conducted in Norway), as well as observation and interviews (conducted in Belgium). The outcome of these studies was a set of social requirements for collaborative applications like Talhonia.

The design of the application followed a participatory design model, where lead users were brought in as active participants in the design process. In an early study, lead users where

presented with various activities intended to gather information on users and different aspects of the process of creating stories collaboratively. Low tech prototyping methods were employed to test out different ideas. For instance, the communication game allowed participants to create stories collaboratively with different levels of "exposure" to other participants, starting of quite isolated (only text is shared via post-it notes delivered by messenger), and becoming a more shared experience towards the end through the use of a telephone conference.

5.3 Study 1 - June study

This section summarizes the first Talhonia study, conducted in June 2008 at SINTEF. The goal of this study was to analyze user experience through the use of the Citizen Media user experience indicators, as well as investigating usability issues in general. Section 5.3.1 summarizes the research method employed, based on the more detailed description given in Lüders et al. (2008). In line with the research goal of this thesis, I have conducted my own analysis of this data with the aim of identifying pieces of reusable knowledge that can be turned into design patterns. The analysis is described in section 5.3.2.

5.3.1 Method

The description of method is based on Lüders, et al. (2008). An integrated approach was employed for this study, focusing not only on usability, but also on several factors related to user experience. These factors have been identified within the Citizen Media project as suitable indicators for measuring user experience. The evaluation method comprised a modified form of the "group-based expert walkthrough". In addition, elements from focus groups are used.

Talhonia is primarily envisioned as a community tool; groups of people that share a common interest can use this tool to write texts collaboratively. In order to evaluate the application in a valid way, it was thus required that the participants shared a common interest in order to simulate the community aspect to a certain degree. To this end, users with an interest in bicycles were recruited from the Norwegian SNS Underskog. The choice of bicycle interest was purely incidental, but the choice of recruiting via Underskog was an intentional move in order to recruit users that had extensive experience with participatory media sites, although they had no specific expertise in usability or storytelling. 12 participants were recruited in total, and were distributed among two evaluation sessions, the first taking place on the 11th of June (with 7 participants), and the next the following day (with 5 participants). Both sessions lasted roughly 150 minutes. The two evaluation sessions were however structured slightly differently. Both evaluation sessions started with a 20 minute introduction to the evaluation and the principles of usability, followed by a 30 minute hands-on collaborative writing period which was included so that users could become familiar with the Talhonia application. In the first evaluation, the collaborative writing session was followed by a 60 minute group-based walkthrough, where usability issues were addressed including a brief wrap-up, and finally focus group discussions took place for about 40 minutes. In the second evaluation these phases were intertwined, allowing the group to pursue discussions naturally as they appeared.

The evaluation sessions were guided by two researchers, one functioning as test-leader, the other as observer. The test-leader interacted with the participants and guided them through the different phases of the evaluation, while the observer took notes and checked whether all relevant issues were addressed. In addition, the whole evaluation session was recorded, and partially transcribed (i.e. notes were taken while playing back the discussions, with particular attention paid to quotes illustrating contrasting views).

5.3.2 Analysis

For my own analysis, I reviewed all the documentation from the study, including the individual evaluator documentation, as well as the test observer documentation. I also listened to the audio recordings of both sessions. This was necessary because I have not used the June version of Talhonia myself, and therefore some of the points in the test documentation were difficult to understand. Listening to the audio recordings helped clear up these points, and it was also necessary in order to check whether all important points that were discussed during the sessions were adequately covered in the existing documentation. During this process the existing documentation was found to be mostly very accurate, but some points were added or elaborated on slightly. The analysis resulted in a written summary (see Appendix B) which became the basis for identifying user experience patterns. The design patterns are presented in the next chapter.

5.4 Study 2 - January study

This section summarizes the second Talhonia study, which was conducted at Ila primary school in January 2009. This study was conducted with a completely different user group, in line with preferences from the developer of Talhonia. The research method employed in this study is described below, based on the description in Wurhofer, et al. (2008).

5.4.1 Method

A total of 35 pupils aged eight years (21 girls, 14 boys) participated in the study, which was conducted over 9 days. 4 pupils participated every day, except for one day where 3 pupils participated. The pupils were given a teddy bear equipped with a Tikitag[10], which functioned as a key for entering Talhonia through the use of a Tikitag reader. They were then presented with the objective, namely to write a story together with another participant. To determine which story the participants should work on, story cards were employed. The story card is a piece of paper with the title of the story and a picture related to the story. The teddy bears given to the pupils represented the characters in these stories. The concept of having story cards and teddy bears was devised in order to make the concept of co-writing more attractive for the pupils. Additionally, this arrangement could potentially make it easier for the pupils to relate to their characters, thus making it easier for them to do the actual writing. Two rooms were utilized and the pupils were arranged so that one pupil from each pair sat in the same room. This arrangement hindered the pupils working on the same story from communicating directly with each other, thus they would need to do all communication from within Talhonia. One or two researchers were present at all times to observe the pupils, as well as assisting

[10] Tikitags are based on RFID technology, and allows RFID tags to be used to trigger various actions on a computer, such as opening a web page.

them with any technical difficulties. The actual writing and generation of ideas was left completely to the pupils. The co-writing sessions lasted for 30-45 minutes. After the co-writing sessions, the pupils participated in a group interview with the main researcher. The interviews focused on the pupils' experiences related to the co-writing process. They were also asked about reading and computing habits in more general terms, including whether they wrote on the computer in other settings as well. The group interviews typically lasted between 10 to 15 minutes and were taped for later analysis.

5.4.2 Analysis

As in the previous study, I conducted my own analysis of this study, with the aim of identifying user experience patterns. The analysis is based on my own notes and observations[11], a summary document from the main researcher on the project, as well as the recordings of the interview sessions. The analysis of these materials resulted in a written summary which formed the basis for identifying design patterns (see Appendix C). In the first Talhonia study, the functionality for synchronous co-writing was not yet implemented, so with respect to the current analysis, the most important aspect of the second study was to see how synchronous co-writing worked in practice.

Having children as evaluators proved to be a different experience from using adult evaluators, the main difference being that the children were much more forgiving of the idiosyncrasies of the application, and for the most part found the experience enjoyable despite various problems (both technical and usability-related). Problematic aspects were nonetheless easy to spot by observing the pupils interacting with the application, and as such the present findings should be just as valid as the findings from the June study. The present study had a stronger focus on the actual process of co-writing; where the previous findings often centered on specific usability-related issues, the current findings provide a good starting point for identifying patterns that address co-creative processes.

[11] I attended the study on 3 days, assisting the pupils with technical difficulties and observing them in their interaction with the Talhonia application.

6 UX patterns based on Talhonia studies: Results

This chapter presents the user experience patterns I have identified, based on the analysis of the June and January studies of Talhonia. The patterns adhere to the guidelines for writing patterns identified in chapter 4. Section 6.1 provides a definition of user experience patterns. The patterns are presented in section 6.2, followed by a summary in section 6.3.

6.1 Definition of UX patterns

A user experience pattern is defined in Karahasanovic et al. (2008) as capturing the essence of a successful solution to a recurring problem or demand in a social network system. Furthermore, user experience has been defined as a construct consisting of 8 factors, referred to as UX factors. A collection of user experience patterns should ideally address all of these factors, so that the pattern collection as a whole identifies elements of an application that are needed in order to:

- make the application more fun to use (UX1 Fun/enjoyment)
- evoke positive emotions during use (UX2 Emotion)
- motivate repeated use of the application (UX3 Motivation)
- increase the user's engagement (UX4 Use engagement)
- involve the user (UX5 User involvement)
- promote co-experience (UX6 Co-experience)
- improve sociability (UX7 Sociability)
- improve usability (UX8 Usability)

6.2 UX patterns

Table 7 presents an overview of the patterns I have identified, describing their name, which evaluation study they are based on, and finally which user experience (UX) factors they cover. UX factors are sorted according to relevance, so that the most relevant UX factor is listed first. Following this each pattern is presented separately, preceded by a description of the empirical background of the pattern (i.e. what findings the pattern is based on). Some patterns are followed by a brief discussion of related theory.

Table 7: Overview of identified patterns

No.	Pattern name	Source	UX factor(s) covered
1	Concise Introduction	Talhonia June evaluation	UX8 Usability, UX3 Motivation
2	Gradual Involvement	Talhonia June evaluation	UX3 Motivation, UX8 Usability
3	Gradual Innovation	Talhonia June evaluation	UX4 Use engagement, UX5 User involvement
4	Dynamic Grouping	Talhonia June evaluation	UX7 Sociability, UX6 Co-experience, UX8 Usability
5	Smart Forms	Talhonia June evaluation	UX8 Usability

6	Direct Manipulation	Talhonia June and January evaluations	UX8 Usability, UX1 Fun
7	User-Centered Updates	Talhonia June evaluation	UX7 Sociability, UX5 User involvement, UX8 Usability
8	Version History	Talhonia June evaluation	UX5 User involvement, UX6 Co-Experience
9	Consolidation View	Talhonia June evaluation	UX8 Usability, UX4 Use engagement
10	Idea Room	Talhonia January evaluation	UX6 Co-experience, UX1 Fun
11	Point Of Reference	Talhonia June and January evaluations	UX4 Use engagement, UX1 Fun
12	Quick-Start Template	Talhonia January evaluation	UX4 Use engagement, UX6 Co-experience

6.2.1 Pattern 1 – Concise Introduction

During the Talhonia June evaluation, concerns were raised regarding the layout of the Talhonia login screen. The evaluators were confused by the use of a wood panel texture as a background image, feeling that it gave a misleading impression of the system. In addition, no information was given about the nature of the application. This feedback led to the identification of the current pattern. The next pattern, Gradual Involvement, represents an alternate or complimentary approach for introducing users to a site or application.

1. Concise Introduction

Name:	Concise Introduction (UX8 Usability, UX3 Motivation)
Problem:	How can one help users to quickly understand what the site is about upon first visit?
Forces:	- If users are required to register and log in before they can determine what the site is about, they are likely to leave. - Users who have decided to try the site might become discouraged to do so if the site description is uninteresting and does not meet prior expectations, or is difficult to understand.
Context:	You want to encourage users to try the site.
Solution:	Communicate the main idea of the site as clearly and concisely as possible on the front page (entry point) of the site. If the site targets a specific audience, this should be clearly stated. Employ an illustration and/or text. Text should be no more than two sentences. If the nature of the site is hard to communicate through static text and images, a video can be considered. A video is likely to be less effective however, because it requires deliberate action and resources from the user (i.e. pressing play and spending time watching the video). Applying this pattern is particularly important for sites that don't support Gradual Involvement.
Examples:	Facebook[12] login page: "Facebook helps you connect and share with the people in your life."

[12] Facebook: http://www.facebook.com/ (accessed 3.12.2008)

6.2.2 Pattern 2 – Gradual Involvement

The current pattern represents an alternative as well as a complimentary approach to the findings presented with the previous pattern.

2. Gradual Involvement

Name:	Gradual Involvement (UX3 Motivation, UX8 Usability)
Problem:	How can the threshold for joining the site be reduced?
Forces:	- Users are rarely motivated to go through a registration process if they are unsure about the benefits. - Users need time to build trust towards a site. - Registering on a site requires effort and trust from the user.
Context:	You want to minimize the possibility that users abandon the site.
Solution:	Don't require users to register or complete any kind of task in order to begin using your site. At least part of the site should be available to the user without requiring any action on part of the user. Gently remind the user of the benefits of registering at appropriate places within the system. For instance, if the user is currently trying out a feature which is enhanced when the system knows who the user is (i.e. after registration), this is a good time to remind the user about the benefits of registering. However, reminders should be toned down, and not appear too often (once per session should be ok), or users might find them irritating. This pattern is also applicable to sites employing several levels of registration. For instance, if the user has gone through an initial registration, there might be features of the site that works better when the user has submitted additional information, such as interests, musical tastes, and so on. The same approach can be used in these cases by reminding the user at the appropriate place. To accommodate registered users who find such reminders irritating, these types of reminders can be presented together with an option for toggling them off. If this solution is considered unsuitable, Concise Introduction can be considered instead.
Examples:	MySpace[13] allows unregistered users to browse profiles, watch videos, and listen to music, but in order to add their own content or comment on other peoples' profiles, they need to register. This represents a way of gradually involving the user by showing him the added benefits of registering.

[13] MySpace: http://www.myspace.com/ (accessed 29.11.2008)

6.2.3 Pattern 3 – Gradual Innovation

Talhonia represents a novel idea for a web application. This issue was discussed during the Talhonia June evaluation, and the evaluators requested features similar to those they were used to from social network sites such as Facebook. Among other things, they wanted to be able to form groups for people interested in writing about different topics, and access to newsfeeds informing them about the latest changes (i.e. which stories have been edited since the user was last logged in).

3. Gradual Innovation

Name:	Gradual Innovation (UX4 Use engagement, UX5 User involvement)
Problem:	How can one help users to embrace new and unique network application concepts?
Forces:	- If a service feels too familiar, users will not be motivated to try it because it doesn't offer anything new. - If the service feels too different, users will have a hard time understanding and relating to it, and thus not feel motivated to spend time on it.
Context:	You are building a social network application around a novel/innovative concept.
Solution:	Blend the innovative concept with traditional SNS elements. Traditional elements that people expect to see in SNS's include user profiles, ability to add other users as friends, and possibilities for communicating with other users. In addition, most social network applications present users with news about friends of the user or groups the user might be associated with. This helps keep people informed about what their friends are doing and supports a feeling of belonging to a community. See the User-Centered Updates pattern for more on this.
Examples:	Most successful SNS's have one or more "innovative" features which are combined with social features, such as YouTube[14] which makes watching videos a more social experience.

Related theory

This pattern is in line with the work by Boyd and Ellison (2008), who identified typical elements of SNS's. These include the provision of user profiles, the ability to add other users as friends, and possibilities for sending messages and comments to other users.

[14] YouTube: http://www.youtube.com/ (accessed 5.12.2008)

6.2.4 Pattern 4 – Dynamic Grouping

When discussing the June version of Talhonia, the evaluators wondered how the stories created by users could be organized so that each user could manage to keep up with the type of stories that interested him. Possible solutions to this problem were suggested, and include features such as providing support for friends and/or groups which can serve as a basis for filtering information. For instance, a typical implementation might filter information so that only information related to the user's friends or groups is presented.

The evaluators felt that navigation in Talhonia quickly became troublesome when there were many stories to choose from. As the user base grows, information overload might become a problem for the users. One strategy is to use rights management, so that only users who have access to modify or read a story will be able to view it. An active user might nonetheless participate in a large number of stories, which will quickly become unmanageable if this information is not filtered in some way.

4. Dynamic Grouping

Name:	Dynamic grouping (UX7 Sociability, UX6 Co-experience, UX8 Usability)
Problem:	How can one avoid overwhelming users with information?
Forces:	- If too much information is displayed, people might loose touch with their contacts and find the service overwhelming. - If too little information is displayed, people might loose interest and feel that there's not enough happening.
Context:	The user is overloaded with updates pertaining to content generated by other users.
Solution:	Firstly, find ways to filter the information, for instance according to interests, friends, or groups that the user might be associated with. Only information that is relevant to the user's friends, groups, or interests should be presented to a user. If there are a large number of users, and users typically have large numbers of friends or contacts, it might be more viable to let the user decide what types of information should be sent for different types of friends. Secondly, information which concerns the same content or the same people should be grouped together where this is natural. This lessens the number of distinct information pieces that are visible, and helps the user to stay on top of the latest happenings. For instance, group news items that are related to the same content item. Instead of presenting four events related to a content item as four separate news items, these can be grouped to form one larger news item. This will occupy less space in the user's attention space. To implement this, one could construct rules that govern the chunking of information. For instance, if an event related to a content item

	occurs less than 24 hours after the preceding one, the original news item should be updated to reflect this, rather than resulting in another news item being generated.
Examples:	Newsfeeds on MySpace and Facebook filter information according to the users' friends, and Facebook groups events occurring within a certain timeframe.

Related theory

The amount of user-generated content available on a site can quickly grow large as the use of a site increases[15]. One way of combating the increasing clutter as use of the service increases is to utilize the concept of chunking. Within cognitive psychology, a popular theory of information processing asserts that humans deal with large amounts of information by grouping isolated pieces of information into semantically meaningful chunks of information (Miller, 1956). Chunking of information is expected to reduce cognitive processing overhead for the user, and thus make the processing of the information more efficient. This general information processing strategy can be exploited in applications like Talhonia as well. For instance, if a story has been edited several times during a short period of time, this information could be presented together as one chunk of information (i.e. one news item), rather than a long list of separate news items which is more likely to confuse the user.

[15] YouTube is a case in point; according to estimates from Digital Ethnography (YouTube statistics: http://ksudigg.wetpaint.com/page/YouTube+Statistics?t=anon, accessed 26.02.2009), over 200,000 videos are uploaded every day.

6.2.5 Pattern 5 – Smart Forms

The current pattern resulted from some usability issues that were raised in the Talhonia June evaluation, and focuses on how to make a system more effective and pleasing for the user:

- In order to add a comment to a text entry in Talhonia users first had to remove instructional text already present in the text field ("Write a comment here…").
- In order to be able to type into the story window, users first had to click on it.

Although these are minor grievances, they seemed to cause a great deal of frustration for the evaluators, possibly because they occurred in functionality that is used frequently.

5. Smart Forms

Name:	Smart Forms (UX8 Usability)
Problem:	How can the inputting of data be optimized to avoid fatiguing users?
Forces:	- Users quickly tire of tasks that feel redundant or are needlessly complicated to perform.
Context:	Users need to be able to input data quickly and effectively.
Solution:	If the application employs form fields with instructional text present, such as "*Enter search terms*" or "*Write comment*", make sure that these phrases are deleted automatically when the user clicks on them to enter text. Make text input easier by offering suggestions or predictions based on user input. For instance, remember information the user has entered earlier and use it to make informed suggestions as the user is typing. This applies particularly to searching. If users are presented with possible search suggestions as they type, they can complete the action much more quickly, as well as possibly finding more appropriate search terms as they type. Offer good defaults when applicable. This involves identification of the choices that are made most frequently by users (for instance when selecting from different options in a drop-down menu), and using these as default values. This minimizes the chances of users having to select other options, resulting in an efficient and painless interaction.
Examples:	Modern web browsers such as Mozilla Firefox 3 make extensive use of predictive input.

6.2.6 Pattern 6 – Direct Manipulation

The implementation of the image function in Talhonia received many criticisms during the June evaluation. It was not possible to manipulate pictures in context of the story. Rather, images were displayed in a separate window, with no indication as to where they belong within the context of the story. Furthermore, images could not be resized to fit inside the window.

During the January evaluation, the functionality for editing text in Talhonia was observed to be cumbersome, and made it easy to commit certain types of grammar mistakes. This problem stems from the fact that users cannot enter text directly into the story window. Instead, a separate window appears whenever users want to enter text, splitting the story window into three separate windows showing the preceding, current, and subsequent pieces of text separately. This functionality causes several problems in practice. For instance, when deleting text, one has to select part of the text so that it appears in the edit window. This text must then be deleted, and to commit the change one must click the button entitled "Insert text", which in this context becomes self-contradictory. Inserting text is also problematic. Because the text one enters appears in a separate window, it can be difficult to work out whether spaces need to be added to either side of the text one enters. The result is that words will often end up being spaced inadequately. The stories the children worked on during the Talhonia January study clearly reflect this, and missing spaces between words were particularly prevalent. One of the pupils criticized the editing functionality explicitly, saying that "he had never seen a text editor where you couldn't type directly into the text". The other pupils, while not explicitly criticizing the editing mechanism, nonetheless required much assistance with this function.

6. Direct Manipulation

Name:	Direct Manipulation (UX8 Usability, UX1 Fun)
Problem:	How do you support editing of content in a way that feels natural to the user?
Forces:	- If content editing is cumbersome, little content is likely to be created. - If content is displayed differently between editing and viewing, users might become confused.
Context:	Users need to edit content in a way that feels natural.
Solution:	Avoid unnecessary abstractions that the user has to interpret in order to produce content. For instance, if the user wants to edit a piece of information on the screen, let him manipulate this information directly where it is, rather than opening an editor window somewhere else on the screen. If the content is edited where it appears in a direct manner, the user avoids having to consciously visualize what the end result might be, because the finished result is always visible.
Examples:	WYSIWYG[16] editors such as the one in Microsoft Word.

[16] WYSIWYG: What You See Is What You Get

6.2.7 Pattern 7 – User-Centered Updates

A feature which lets users see recent changes to a document was requested several times during the Talhonia June evaluation. More specifically, the evaluators wanted to be able to see at a glance which parts of the document had recently been edited, and by whom. Several evaluators stressed that this feature is essential. This concern might be less important in the January version of Talhonia, where the focus is primarily on synchronous collaboration. However, the current pattern should be considered for any application where asynchronous collaboration is supported.

7. User-Centered Updates

Name:	User-Centered Updates (UX7 Sociability, UX5 User involvement, UX8 Usability)
Problem:	How can one help users to stay up-to-date on collaboratively created content?
Forces:	Users are unlikely to partake in collaborative efforts if it is difficult to stay up to date on the changes made by others.
Context:	Users are working collaboratively on a content item and need to stay updated on the latest changes.
Solution:	Help users stay updated on recent changes by highlighting what has changed in the period since the last login. Rather than simply presenting the information that is available, show it in a way which highlights changes that are relevant to the user. If a content item the user is interested in has been updated since the user last logged in, it would be reasonable to present an excerpt of what has changed, so that the user doesn't have to look for this information manually. See Dynamic Grouping for more on how information about updates can be presented to the user.
Examples:	Newsfeeds on Facebook and MySpace.

Related theory

The human perceptual system is particularly sensitive to detecting changes in our environment. Presenting information in terms of what has changed thus feels natural to us (Woods, Patterson, & Roth, 2002). When collaborating on content asynchronously, presenting users with the latest changes can spark creativity and thus help the collaborative process.

6.2.8 Pattern 8 – Version History

During the Talhonia June evaluation, the evaluators discussed the version history function in Talhonia. The current implementation, which only allows viewing earlier versions of a document, was felt to be too limited. In order to serve a useful purpose, functionality for reverting to an earlier document version was considered essential.

8. Version History

Name:	Version History (UX5 User involvement, UX6 Co-Experience)
Problem:	How can participants edit content in a collaborative process without invalidating the work of other participants?
Forces:	- Users might be reluctant to contribute if they feel that their content can be removed easily by others. - Users might be reluctant to contribute if they feel that they may easily invalidate the work of others.
Context:	Participants need to be able to modify and create new content in a co-created item without having to worry about destroying other participants' work.
Solution:	Every time a document is edited, a new version should be created. Display a list of these versions next to the currently open document, and enable users to navigate through the various versions. In addition, it should be possible to revert back to an earlier version of the document. Since new versions of a document are added to the history list on every edit, one might consider incorporating principles from Dynamic Grouping to avoid unnecessary clutter. The functionality of the version history list can be expanded with the Consolidation View pattern.
Examples:	The version history function in Talhonia, although this implementation is currently only an incomplete implementation of this pattern.

6.2.9 Pattern 9 – Consolidation View

Although the provision of a revertible version history function was the main feature evaluators missed in the June evaluation, one evaluator also asked whether it was possible to copy pieces of text that was in the document at an earlier stage into the current document version. This is currently only possible via manually copying and pasting text. For larger documents, such a task will quickly become unmanageable, since the user will be forced to jump back and forth between different document-versions to copy and paste text. Furthermore, new versions of the document will be created every time a new piece of text is pasted in, thus making it increasingly more difficult for the user to maintain an overview of the versions available.

9. Consolidation View

Name:	Consolidation View (UX8 Usability, UX4 Use engagement)
Problem:	How can users easily consolidate different versions of a document?
Forces:	- In a collaborative process where each participant adds and edits content at whim, pieces of content will gradually get "buried" as new versions are created.
Context:	In an application that supports a version history, users need a simple way of consolidating pieces of text from different versions of a document.
Solution:	Implement a special editing view which allows participants to view two (freely selectable) versions of a document side by side. It should be possible to easily move content between documents, for instance by means of dragging text from one document to another. When this view is closed, the revised document should be saved as a new version of the current document.
Examples:	"Compare documents" function in Microsoft Word.

6.2.10 Pattern 10 – Idea Room

Since Talhonia does not support a mechanism for communication among users, the pupils found their own ways of communicating, either by walking to the other room and talking with their partner, or by using the story window in Talhonia as a sort of primitive instant messaging tool. These observations strongly suggest that a means of communication with co-writing partners is needed in Talhonia, and led to the identification of the current pattern, which emphasizes the need for a space where participants can meet to share and discuss ideas during content creation.

10. Idea Room

Name:	Idea Room (UX6 Co-experience, UX1 Fun)
Problem:	How can participants in a collaborative project communicate with each other from within the application?
Forces:	- If participants can't communicate with each other in real time when creating or editing collaborative content, collaboration will be difficult.
Context:	Participants need to communicate ideas related to their collaborative project.
Solution:	Support an idea room; a part of the system where participants in a project can meet and discuss ideas relating to their project. It can be implemented via instant messaging, video conferencing, or audio conferencing. In addition to this feature, an idea room could feature a way of extracting specific ideas exchanged in discussion and pin it on an idea board. Such an idea board can function as an overall guideline for the participants working on a project.
Examples:	Instant messaging and audio/video-conferencing tools such as Windows Live Messenger and Skype.

6.2.11 Pattern 11 – Point Of Reference

During the January evaluation of Talhonia, a concept was tested where each pupil was assigned their own teddy bear, which would serve as a physical representation of the character they would be writing about. Thus, the teddy bear shared the same name as the corresponding character in the story. The teddy bears were equipped with small transmitters called Tikitags, which were used together with a Tikitag reader to provide an entry-point into Talhonia; when the teddy bear was placed on top of the Tikitag reader (which was connected to the computer via a USB connection), Talhonia would be launched automatically. The pupils were observed to perform various gestures towards their teddy bear during the writing session, indicating that they connected in some way with their teddy bear while writing.

In the Talhonia June evaluation, the evaluators discussed the use of maps as a desired feature in Talhonia, as this would allow them to tell a story (in this case about a bike trip), related to a particular location, or sets of locations. This idea is conceptually very similar to the use of teddy bears as a physical representation of a character, the difference being that maps and teddy bears are useful in different writing situations. More generally, the provision of a point of reference of some kind seems to be a useful aid when co-writing stories.

11. Point Of Reference

Name:	Point Of Reference (UX4 Use engagement, UX1 Fun)
Problem:	How can one help users co-writing stories to maintain a basic structure to the output of each participant?
Forces:	- If users are working co-creatively and don't have any rules which help structure each individuals output, the end result will usually end up cluttered and incoherent. - If there are too many rules guiding the co-creative process, it can hamper the creativity of the participants and negatively affect the end result.
Context:	Users co-writing stories need a way of structuring the creative process in order to avoid overlapping and incoherent output.
Solution:	Provide a point of reference for each participant in a co-creative project. A point of reference is intended to make it easier for participants to divide responsibilities when writing a story, so that each participant writes something that is related to a particular aspect, like an object, a character, or a location. Different types of reference points will be suitable depending on the nature of the content that is created. If it is meaningful to relate the story to geographic locations, a particular location on a map could be used as a point of reference (e.g. for adding content related to a holiday trip, or thoughts about a particular shop or restaurant, or anything else that can be shown on a map). Having each participant write something that relates to a specific location will reduce the possibility of overlapping content.

	For character-based story-writing, having representations of the character one is writing about can be helpful, particularly for younger users. This will help them focus their writing on a particular character.
Examples:	In the January evaluation of Talhonia, the pupils were given teddy bears that represented the characters they were writing about. This helped them to focus on their character in the story.

6.2.12 Pattern 12 – Quick-Start Template

During the Talhonia January evaluation, story cards were used to present the theme of the story the pupils were writing, as well as the main characters involved. The main characters corresponded to a physical teddy bear, which the pupils would be given as part of the preparation for the co-writing process. The story cards were used as a device for assigning the pupils to a story, which also decided which teddy bears they were going to write about. Although the pupils seemed to have favorites among the teddy bears, none complained when they were assigned another one.

The amount of time it took for a pupil to get comfortable and start writing differed greatly, but most pupils got started quickly (within 2 minutes) and almost all participants had contributed to the project by the end of the writing session (about 30 minutes).

12. Quick-Start Template

Name:	Quick-Start Template (UX4 Use engagement, UX6 Co-experience)
Problem:	How can one help users co-writing stories to decide on a common starting point or theme for the story?
Forces:	- Group discussions on issues where there are no hard answers can quickly escalate into long-lasting debates. - Predetermining the goal of a collaborative process can be limiting and compromise the creative output.
Context:	Users co-writing stories want to get started quickly.
Solution:	Provide content templates that provide a foundation for the form of the content, by specifying the title, as well as main themes and characters and their relation. The advantage of this approach is that it provides a neutral starting point for the collaboration, helping the participants get involved in the actual creative process almost instantly. It can also function as a source of inspiration for the participants. For more flexibility, content templates could be presented several at a time, allowing each participant to vote for the one they want to use. This will allow the group some freedom in selecting a template, without risking long-lasting discussions, since a purely quantitative voting approach should be considered fair by all participants. The concept of templates can be applied to many different contexts, such as meeting reports (a template with a general structure), holiday planning (a list prefilled with typical items which the participants can then edit), and so on.
Examples:	In the January evaluation of Talhonia, the pupils were presented with story cards which specified the theme of the story by means of a descriptive title which also specified the main characters of the story. This helped them to get started writing quickly

6.3 Summary

In this research 12 user experience patterns have been identified based on findings from two evaluation studies of Talhonia. The findings and resulting patterns are summarized below. Finally, their coverage according to the user experience factors is assessed and briefly discussed.

The evaluators found it difficult to understand what the June version of Talhonia was about upon first use. Based on this feedback, the Concise Introduction and Gradual Involvement patterns were identified. These two patterns represent alternate and complementary approaches to introducing users to a web-based application.

The Gradual Innovation pattern resulted from several comments about the June version of Talhonia lacking a class of functionality that was deemed important or necessary, such as making it possible to form groups with other users, as well as functionality for receiving news on updated content. This type of functionality represents something close to a convention in social network applications. Gradual Innovation thus focuses on the importance of implementing such features even though the application in question might be conceptually different from existing social network applications (as is the case with Talhonia). This pattern refers to the User-Centered Updates pattern, which deals with how users can be updated on recent activities, particularly related to co-creative processes. This pattern further refers to the Dynamic Grouping pattern, which is a lower level pattern dealing with how news updates can be communicated to users without taking up too much of their attention space (i.e. causing information overload).

The Smart Forms pattern builds on feedback related to usability from the Talhonia June evaluation, focusing on the importance of streamlining data input and interaction with form fields.

There were difficulties related to content editing in both the June and January evaluations of Talhonia. In the June evaluation, these difficulties were related to the functionality for adding images to a story, and in the January evaluation, they were related to the functionality for inserting text into a story. These findings led to the identification of the Direct Manipulation pattern.

In the Talhonia June evaluation, the version history was criticized for only allowing viewing of different versions, meaning that it was hard to use for any practical purpose during content creation. Based on this input, the Version History pattern was identified, presenting the concept of a version history and incorporating the criticisms leveled at it. Comments from one evaluator further helped the identification of a related pattern, Consolidation View, which focuses specifically on how a story can be managed across different versions of a document.

Observations from the Talhonia January evaluation strongly indicate that a means of communicating live with co-writing partners is needed in Talhonia. This led to the identification of the Idea Room pattern. Experiences from this evaluation also led to the identification of two additional patterns related to co-writing stories. Point Of Reference concerns how the provision of a common point of reference, such as a physical character or a

map, might be helpful when co-writing stories. In a similar vein, Quick-Start Template addresses a way of kick starting a co-writing session by providing templates or short story-ideas which can be used as starting points.

Table 8 presents an overview of the coverage of user experience factors for the patterns presented in this chapter. This clearly shows that UX8 Usability is overrepresented, while UX2 Emotion is not covered in any pattern. The reasons for this skewed distribution could be many, but one possibility is that it reflects the general distribution of findings in the Talhonia studies (i.e. the evaluators focused more on usability than emotion for instance). Also, it's likely that some of the factors are harder to measure and interpret from findings. In this regard, emotion is probably one of the most difficult factors to measure in a valid way. Another contributing factor to the skewed distribution might be that the UX factors are not very clearly defined (see section 6.1 for the definitions). For example, UX3 Motivation, UX4 Use engagement, and UX5 User involvement are all overlapping terms. The same can be said for UX1 Fun/enjoyment and UX2 Emotion, particularly as UX2 Emotion is defined to signify positive emotions.

Table 8: Coverage of user experience factors in patterns identified from Talhonia studies

	Concise Introduction	Gradual Involvement	Gradual Innovation	Dynamic Grouping	Smart Forms	Direct Manipulation	User-Centered Updates	Version History	Consolidation View	Idea Room	Point Of Reference	Quick-Start Template
UX1 Fun/enjoyment						•				•	•	
UX2 Emotion												
UX3 Motivation	•	•										
UX4 Use engagement			•						•		•	•
UX5 User involvement			•				•	•				
UX6 Co-experience				•				•		•		•
UX7 Sociability				•			•					
UX8 Usability	•	•		•	•	•	•		•			

7 Merging SINTEF UX patterns

The patterns identified within the Citizen Media project are meant to be presented as a collection of patterns. To this end, it's a natural next step to merge new patterns with those identified previously, so that a coherent collection of patterns is formed. In this chapter I will merge the user experience patterns that have been identified at SINTEF during an evaluation of the Urørt application[17] with the ones that have been identified in the present research. The strategy for pattern merging is described in section 7.1. Section 7.2 describes the assessment of patterns, and the revised patterns are given in section 7.3. Section 7.4 provides an overview of the complete set of patterns as they appear after the merging has been conducted. Following this, section 7.5 discusses the patterns in light of commonly known pattern collections within HCI. Finally, section 7.6 summarizes the lessons learned from identifying and writing patterns.

7.1 Strategy for pattern merging

In order to improve usability of a pattern collection, it's important that the patterns remain consistent. A prerequisite for consistency is that all patterns conform to a common definition. The guidelines for pattern writing presented in Table 5 (in section 4.1) formed the basis for analyzing the first set of Citizen Media user experience patterns, including the patterns that were identified from the evaluation of the Urørt application. The problems that were found during this analysis resulted in a list of recommendations for improvement. To ensure consistency with the new set of patterns, the first set of patterns should be revised according to these recommendations. Table 10 provides a summary of the pattern writing guidelines (see Table 5 in section 4.1 for a complete description).

Table 9: Summary of guidelines for pattern writing

Section	Guideline
Name	Patterns should be named after the feature or artifact that results from applying it. Furthermore, names should be kept as short as possible, and the maximum length is 4 words. Names should be consistent across patterns, and ideally be a unique phrase so that they are easier to remember.
Problem	The problem should state the basic problem that is addressed, and it should be framed as a question.
Forces	The forces should describe various trade offs or concerns related to the implementation of the solution presented. The forces should not contain parts of the solution, or describe the context of the pattern.
Context	The context should describe when it is appropriate to apply the pattern.
Solution	The solution should be described concretely so that it can be implemented without raising further questions. If a solution can not be described

[17] Urørt is a social network site for unsigned artists in Norway, and is owned and operated by the public service provider NRK. This site has many similarities with typical Citizen Media applications, and was used in early evaluations before any actual Citizen Media applications were available.

	adequately in a single pattern, it should reference other sub-patterns that describe these issues separately.
Examples	The examples should support the solution by referring to commonly known implementations of the pattern.

Another point related to consistency is the relationship between the patterns in a collection. Patterns should not overlap in scope or purpose (Vlissides, 1996). Rather, they should present distinct solutions that complement each other. If the patterns in a collection present similar solutions, or solutions that partially overlap with others, it will most likely confuse the reader and the patterns will be less useful. It's therefore important that patterns give a solution to a specific problem, and that any solution is given only once in the pattern collection. If there is a need for patterns that address solutions to multiple issues (as might often be the case for larger-scale patterns), this should be solved by referring to sub-patterns by name rather than repeating the solution itself. This avoids confusing inconsistencies, and it makes the pattern collection easier to maintain, since changes to a specific solution only needs to be made in one place. The process of merging thus implies two steps - ensuring that the patterns do not overlap, as well as ensuring that the patterns are written consistently. The strategy for merging is described in Table 10.

Table 10: Strategy for merging UX patterns

Step	Description
1	If two patterns present practically identical (overlapping) solutions, they should be merged. When merging, the most relevant elements of both patterns are selected and combined.
2	If two patterns present partially overlapping solutions, the pattern should be split so that the overlapping part of the solution is presented as a separate pattern. The relevant part of the solution should then be replaced with a reference. When splitting patterns, step 1 should be repeated so that potential overlap with already existing patterns is avoided.
3	To ensure consistency, patterns from the Urørt evaluation should be revised according to the guidelines presented in Table 9.

7.2 Pattern assessment

In the following, assessments of each of the patterns identified from the evaluation of the Urørt application are presented. This forms the basis for the merged and revised patterns that are presented in section 7.3. An overview of the patterns is given in Table 11. For a complete description of these patterns, see Appendix A (patterns 21 to 26).

Table 11: Overview of user experience patterns prior to merging and revision

No.	UX pattern name
21	Getting in touch with people sharing the same interests
22	Motivating people to make their own music available to others
23	Providing comprehensive information about a networked A/V application on its front page
24	Navigation within a networked A/V application
25	Personalization of user profiles
26	Keeping users updated

7.2.1 Pattern #21 (Getting in touch with people sharing the same interests)

This pattern addresses the issue of communication among users in a networked application. More specifically, it suggests the implementation of public (guestbook, wall) and private (private messages) messaging systems. The aim is to enable human-human interaction, as well as for giving other users feedback on their work. The Idea Room pattern also addresses this issue, but here the focus is on live communication such as chat and videoconferencing. Although these patterns both relate to communication, they address different types of communication and should thus remain separate.

7.2.2 Pattern #22 (Motivating people to make their own music available to others)

This pattern addresses various aspects related to motivating people to share their music with others. These include the ability to upload and promote music, give and receive feedback (general comments, reviews, and ratings), and the ability to easily find new and interesting music. The solution given here is ambiguous, since the solution focuses on three aspects. In addition, these aspects are not presented in adequate detail. To ensure that these aspects are addressed properly, they should be split into several patterns that each targets one aspect (and references the others). The aspects are:

- A: Giving and receiving feedback
- B: Finding new music
- C: Uploading and promoting music

Looking at this list, we find that there is some overlap with existing patterns. Aspect A is addressed in pattern #21 (Communicating with other users), and aspect B is addressed in pattern #24 (Navigation in an A/V application). With some minor modifications, it should thus be possible to cover these two aspects with existing patterns. This leaves aspect C. The pattern should thus be rewritten so as to deal more specifically with this aspect.

7.2.3 Pattern #23 (Providing comprehensive information about a networked A/V application on its front page)

This pattern addresses how the front page of the application should be designed so as to clearly communicate the concept and content of the site. However, it also mentions some

issues related to navigational concepts and the use of tags and genres as structural entities, which makes the solution ambiguous. These additional concerns should be factored out and dealt with separately. There is already a pattern addressing the use of tags for navigation (pattern 24). Furthermore, the core solution of this pattern has a large degree of overlap with the Concise Introduction pattern. Therefore, this pattern should be merged with Concise Introduction.

7.2.4 Pattern #24 (Navigation within a networked A/V application)

This pattern targets navigation mechanisms in general, and is not sufficiently specific. For instance, there are many navigation mechanisms one can implement to improve navigation, such as bread crumbs (Van Welie, 2008), and the possibility of returning to a safe place (Tidwell, 1999). This pattern is particularly focused on the use of tags for supporting navigation, and should thus be rewritten with a focus on this specific navigational aid.

7.2.5 Pattern #25 (Personalization of user profiles)

This pattern addresses the issue of increasing user involvement and motivation by allowing users to personalize their profiles. There is no ambiguity or overlap with other patterns.

7.2.6 Pattern #26 (Keeping users updated)

This pattern addresses the need for mechanisms in the application that help the user stay updated on content updates. This pattern overlaps with User-Centered Updates, and these two patterns should thus be merged.

7.3 Revised and merged patterns

The revised and merged patterns are presented in this section. An overview of the revised patterns, presenting the names of the patterns before and after the merging process is shown in Table 12.

Table 12: Overview of revised and merged patterns

Before merging and revision	After merging and revision
Getting in touch with people sharing the same interests	Private And Public Communication
Motivating people to make their own music available to others	Creation Motivation
Providing comprehensive information about a networked A/V application on its front page	*Merged with existing pattern:* Concise Introduction
Navigation within a networked A/V application	Tag-Based Navigation
Personalization of user profiles	Personalized User Profiles
Keeping users updated	*Merged with existing pattern:* User-Centered Updates

Private And Public Communication (UX5 User involvement, UX6 Co-Experience, UX7 Sociability)

Name:	Private And Public Communication (UX5 User involvement, UX6 Co-Experience, UX7 Sociability)
Problem:	How can communication between people sharing the same interests be supported through a networked A/V application?
Forces:	- Users want to comment on content uploaded by others, and receive feedback on their own content. - Users don't always want their comments to be publicly available. - Users want to initiate contact with other users.
Context:	Participants need to be able to get in touch with other users.
Solution:	A networked A/V application should include as its integrated parts both public and private messaging systems. A public messaging system is often implemented as a guest book, where users can leave comments that are visible to everyone. A private messaging system is often implemented similarly to e-mail functionality, where you can send and receive messages to and from specific users. Providing messaging systems allows simple human-human interaction and lets users provide feedback on each others content.
Examples:	- The wall in Facebook (public communication) and the "send message" function (private communication). - Getting in touch with people sharing the same interests in music through Urørt application, for example contacting potential new band members, organising joint-concerts and play sessions

Creation Motivation (UX3 Motivation, UX5 User Involvement, UX6 Co-experience)

Name:	Creation Motivation (UX3 Motivation, UX5 User Involvement, UX6 Co-experience)
Problem:	How can users be motivated to share their own content?
Forces:	- Users want to receive constructive feedback on their content. - Users want to provide feedback on content from other users.
Context:	Users of a networked A/V application need motivation to publish their own content.
Solution:	Provide functionality that enables users to easily upload their content. The possibility of receiving feedback from other users represents a motivating factor. This pattern should thus be applied in combination with the Private And Public Communication pattern. In addition, it should be possible for users to promote their content, for instance by having the option of informing all friend users when new content is uploaded.
Examples:	Functionality for uploading music and receiving feedback from listeners in the Urørt application.

Concise Introduction (UX8 Usability)

Name:	Concise Introduction (UX8 Usability)
Problem:	How can one help the users to quickly understand what the site is about upon first visit?
Forces:	- If users have to register and log in before they can determine what the site is about, they are likely to leave. - Users who have decided to try the site might become discouraged to do so if the site description is uninteresting and does not meet prior expectations, or is difficult to understand.
Context:	First-time users need comprehensive information about a site without overwhelming or irrelevant content.
Solution:	Communicate the main idea of the site as clearly and concisely as possible on the front page (entry point) of the site. If the site targets a specific audience, this should be clearly stated. Employ an illustration and/or text. Text should be no more than two sentences. If the nature of the site is hard to communicate through static text and images, a video can be considered. A video is likely to be less effective however, because it requires deliberate action and resources from the user (i.e. pressing play and spending time watching the video). Applying this pattern is particularly important for sites that don't support Gradual Involvement.
Examples:	Facebook login page: "Facebook helps you connect and share with the people in your life."

Tag-Based Navigation (UX8 Usability)

Name:	Tag-Based Navigation (UX8 Usability)
Problem:	How can one help users find relevant content within a networked A/V application?
Forces:	- Finding interesting content can be difficult and time-consuming.
Context:	Users need to be able to find relevant content quickly.
Solution:	Make it possible for users to tag their content, and support navigation by using tag clouds. Tag clouds complement regular search functions and make it easy for users to find content that fit into specific categories or contain information related to specific topics. As some users are not familiar with tags, a "what are tags"-link describing this concept should be provided.
Examples:	Navigation mechanisms of the Urørt application.

Personalized User Profiles (UX3 Motivation, UX8 Usability)

Name:	Personalized User Profiles (UX3 Motivation, UX8 Usability)
Problem:	How can one enable users to change the appearance of their user profiles?
Forces:	- Users might not feel ownership of their user profile if they are unable to change it in any way. - Users might find it daunting if there are too many options for changing their profile.
Context:	Users want to have some control over how they are presented in a networked A/V application.
Solution:	The application should provide simple functionality for editing user profiles allowing users to add a personal look to their profile. Functionality should include the ability to change background textures and colors, images related to their profile, as well as the fonts used. More options could be provided, but the number of options must be balanced according to user needs and competence. Common words should be employed to the greatest extent possible, instead of technical terms (e.g. "widget"). Automatic confirmation should be given when a change has been made (e.g. "Your photo has been uploaded"). Enabling users to change the layout and look of their profiles supports a sense of ownership to the application, and can in turn motivate them to spend time on the service. In addition, this feature represents a way for users to enhance usability. For example, users with poor vision can increase font size and use high contrast color schemes.
Examples:	The functionality for personalizing user profiles in the Urørt application.

User-Centered Updates (UX5 User involvement, UX7 Sociability, UX8 Usability)

Name:	User-Centered Updates (UX5 User involvement, UX7 Sociability, UX8 Usability)
Problem:	How can one help users to stay up-to-date on new content?
Forces:	- Users are unlikely to partake in collaborative efforts if it is difficult to stay up to date on the changes made by others. - Users want to receive/provide comments on newly updated content. - Users want automatic news updates from friends.
Context:	Users need to stay updated on what content is added at a site.
Solution:	The application should provide an easy way of sharing information about updates and new content, by enabling users to subscribe to automatic news-feeds from other users. Information in news feeds should be presented so that changes that have occurred since the last login are highlighted. For instance, if a content item the user is interested in has been updated since the user last logged in, it would be reasonable to present an excerpt of what has changed, so that the user doesn't have to look for this information manually.
Examples:	Newsfeeds on Facebook and MySpace.

7.4 Summary

This section provides an overview of the complete set of patterns as they appear after the merging and revision has been conducted (section 7.4.1), as well as a possible visual organization of these patterns (section 7.4.2).

7.4.1 Pattern listing

Table 13 lists the final set of patterns that were identified in this research, including which studies they were based on.

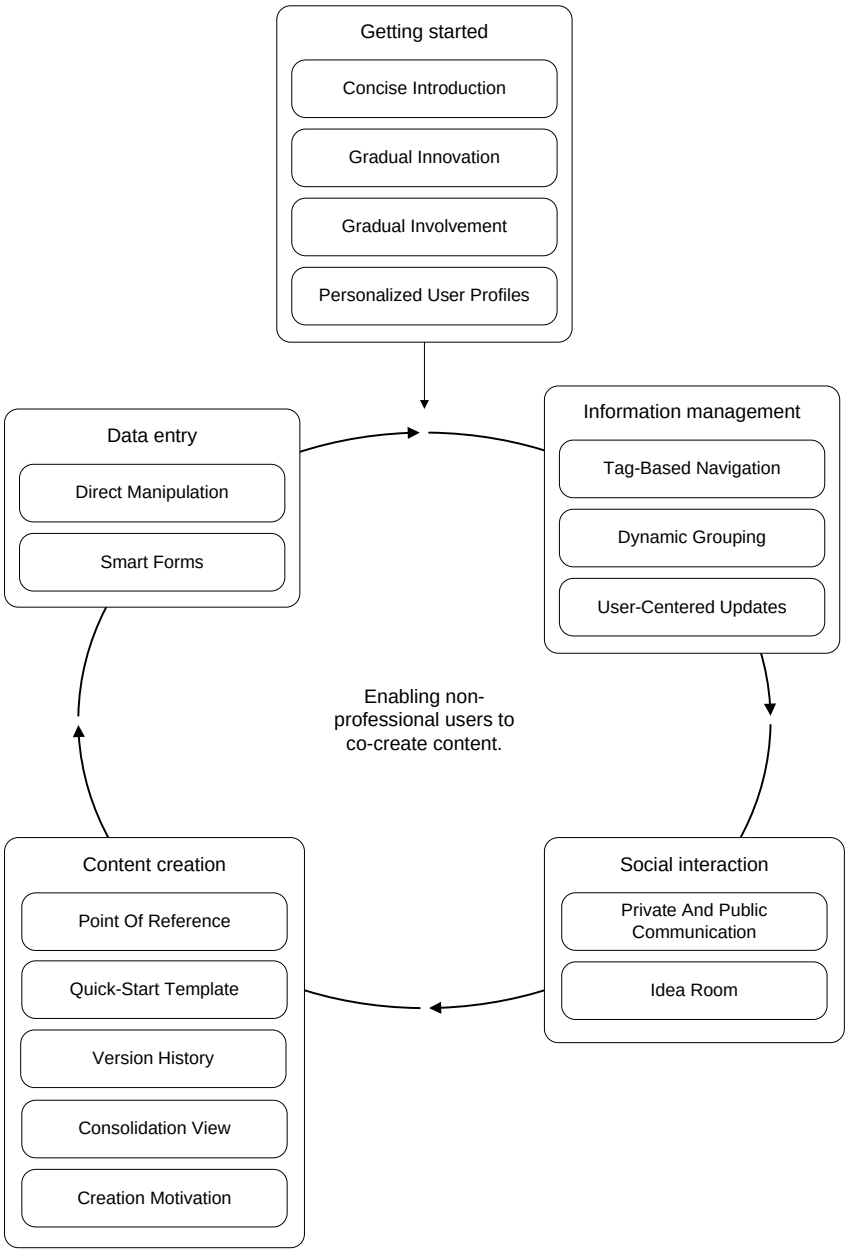
Table 13: Summary listing final set of patterns and which study they are based on

No.	User experience pattern name	Source
1	Concise Introduction	Talhonia June, Urørt
2	Gradual Involvement	Talhonia June
3	Gradual Innovation	Talhonia June
4	Dynamic Grouping	Talhonia June
5	Smart Forms	Talhonia June
6	Direct Manipulation	Talhonia June, Talhonia January
7	User-Centered Updates	Talhonia June, Urørt
8	Version History	Talhonia June
9	Consolidation View	Talhonia June
10	Idea Room	Talhonia January
11	Point Of Reference	Talhonia January, Talhonia June
12	Quick-Start Template	Talhonia January
13	Private And Public Communication	Urørt
14	Creation Motivation	Urørt
15	Tag-Based Navigation	Urørt
16	Personalized User Profiles	Urørt

7.4.2 Pattern organization

Figure 4 presents a visual map of the patterns that have been identified in this research. Patterns have been grouped according to activity domain, in line with Segerståhl and Jokela's (2006) recommendations.

Figure 4: Map of user experience patterns



Each pattern is related to a specific activity. The looping circle in the center visually enforces the idea that these activities together support the main objective of enabling non-professional users to co-create content. The set of patterns listed on the top function as a portal into the system, and aim to make it easy for the user to get started using the system.

7.5 Discussion

This section presents a comparison of the patterns identified in this research with a selection of already existing pattern collections for interaction design. This enables us to determine whether these patterns add anything new to the HCI design pattern corpus.

Three pattern collections were selected for this comparison: "Common Ground" abbreviated CG (Van Welie, 2008) which contains 53 patterns, "The Interaction Design Pattern Library" abbreviated TIDPL (Van Welie, 2008) which contains 131 patterns, and "The Design of Sites" abbreviated TDOS (Duyne et al., 2006), which contains 107 design patterns[18]. The first two collections have been available for several years, and probably represent the two most well-known design pattern collections for interaction design (Segerståhl & Jokela, 2006). While these two collections are aimed at interaction design in general, and thus can be used to support any kind of user interface design, the third collection was selected because it has more in common with the Talhonia and Urørt patterns, as it is focused on patterns for web site interaction design. It is thus expected that there will be a larger degree of overlap with this collection than the other two. Together, these three pattern collections should provide a representative view of the interaction design patterns that are available today, and thus represent a relatively solid foundation for comparison. If no overlap is found in any of the three collections selected, it indicates that the pattern in question is either unique or not yet in broad use. The comparison is presented in Table 14, where the SINTEF user experience patterns are listed to the left and any overlapping (fully or partially) patterns are shown to the right, listed under the pattern collection it is taken from. A hyphen signifies that no overlapping patterns were found for the pattern in the corresponding collection.

Table 14: Comparison of SINTEF UX patterns with other pattern collections available

SINTEF User Experience Patterns	Common Ground	Interaction Design Pattern Library	The Design Of Sites
Concise Introduction	-	Homepage	Up-Front Value Proposition
Gradual Involvement	-	-	Guest Account
Gradual Innovation	-	-	-
Dynamic Grouping	-	-	-
Smart Forms	Good Defaults	Autocomplete	Predictive Input
Direct Manipulation	WYSIWYG Editor	-	Direct Manipulation
User-Centered Updates	-	-	-
Version History	Interaction History	-	-
Consolidation View	-	-	-
Idea Room	-	-	-
Point Of Reference	-	-	-
Quick-Start Template	-	-	-
Private And Public Communication	-	-	-
Creation Motivation	-	-	-

[18] While both CG and TIDPL are freely accessible on the World Wide Web, the TDOS patterns are published in a book, and are available on the WWW only in abbreviated form. It is the abbreviated version which has been used in this comparison.

Tag-Based Navigation	-	Tag Cloud	-
Personalized User Profiles	User Preferences	-	-

From the table above, we see that 7 patterns were found to have a degree of overlap with already existing patterns, while no overlap was found for 9 patterns. The patterns for which overlap was found are discussed in more detail below.

There are differing views on what is required in order to classify a collection of patterns as a pattern language (Appleton, 1997; Borchers, 2001; Pemberton, 2000; Salingaros, 2000). For clarity, this debate is sidestepped in the following discussion and the term pattern collection is used throughout.

Concise Introduction
The Homepage pattern addresses the same problem with a great deal of overlap, and discusses the same issues such as the use of a sentence (tag-line) describing what the site is about, as well as the use of images and animation. However, the Homepage pattern also covers some additional aspects, detailing all the elements that are needed on the front page of a site, such as a search box, various types of navigation, and a language or country selector. The Up-Front Value Proposition pattern is also similar, but is more focused on effective communication on company web sites.

Gradual Involvement
The Guest Account pattern from TDOS presents a similar solution, but is more targeted towards e-commerce web sites where the goal is to maximize sales. However, the premise is the same for both patterns; users can be put off and possibly leave a web site if registration is required in order to use it.

Smart Forms
This pattern focuses on two aspects that are thought to make data input easier for the user. One is the use of predictive search algorithms, and the other is the use of good defaults. All three pattern collections contain a pattern addressing one of these aspects, but none cover both in the same language or pattern. Good Defaults from CG address the use of good default values in forms, while Autocomplete from TIDPL and Predictive Input from TDOS address the use of predictive algorithms to assist data entry.

Direct Manipulation
Similar patterns are found in two of the collections. Incidentally, this pattern is presented with the exact same name in the TDOS collection. This pattern addresses a different type of direct manipulation than the one described in the SINTEF pattern however. The TDOS pattern stresses the importance of direct control of data containers (i.e. windows) that is typically afforded in desktop GUI applications, but which are less common on the web. The SINTEF pattern on the other hand, focuses on the need to manipulate data directly (i.e. in context of the data set it is part of). The underlying principle is however quite similar, which is that humans always favor doing things directly rather than indirectly. This difference nonetheless

suggests that a different name should be chosen for one of the patterns. The SINTEF pattern could be renamed to something more specific to the act of editing content in context, such as In-Context Editing for instance. The WYSIWYG Editor pattern from the CG collection overlaps with the SINTEF pattern to the extent that they are almost identical. The name of this pattern is however only understandable for those who are familiar with the term WYSIWYG, and even then it could need more specificity.

Version History
The Interaction History pattern from the CG collection addresses the same basic problem, but with a different focus. Both suggest the use of a history which lets the user see what has been done, and allowing the user to revert back to an earlier state. The CG pattern focuses on single-user interactions and is mainly presented as an elaborate undo function (i.e. multi-level undo functionality), while the SINTEF pattern focuses on the same functionality in the context of multi-user interaction and how group collaboration can be inhibited when there is a risk that one user can invalidate the work of others.

Tag-Based Navigation
The Tag Cloud pattern from TIDPL addresses the same issue, but in more detail than the SINTEF pattern. The main focus in the latter is on the benefit of allowing users to tag content and using this as an alternative way of navigating content, while the former focuses mainly on how tag clouds are implemented and what sort of situations they can be used for. The Tag Cloud pattern is more comprehensive and probably has more value for reuse.

Personalized User Profiles
User Preferences address a similar issue, but is more focused on enabling accessibility, such as allowing people to set larger font size, higher contrast colors to support impaired vision, and so on. In a social application, it's important for the user to have some control over the artifact so that it can be used to communicate something about who they are. Thus, the possibility for personalization is a common theme in both patterns, but the goal is different.

7.5.1 Summary
Since there does not seem to exist other collections of patterns that address specific issues related to co-creation of content in online applications, it is unsurprising that many of these patterns have not been identified elsewhere. The patterns that are related to more general usability issues however, mostly overlap with already existing patterns, although the degree of overlap varies. For three of the patterns (Direct Manipulation, Tag-Based Navigation, Concise Introduction), the degree of overlap with other pattern collections is very high, while the remaining four patterns overlap to a lesser extent, addressing similar problems but in a different perspective.

For a pattern user, a situation where multiple patterns with different names address identical problems is very confusing and should be avoided. Therefore, patterns that overlap fully with already existing patterns should be discarded and replaced with references to the previously existing patterns. Discarding patterns is a natural part of the pattern development process, and is similar to the general scientific process of discarding less effective theories with more

effective ones. Even though the pattern itself is not useful, discarding a pattern indicates that it is sufficiently specific in order to be falsifiable (or in this case, shown to already exist). This is an important attribute of patterns according to McGee (2007). When overlap between patterns are found, this also implies that a similar problem has been observed earlier, thus supporting the premise from which the pattern is built on.

7.6 Lessons learned

This section summarizes the lessons learned from identifying and writing patterns. Sections 7.6.1 through 7.6.3 discuss issues related to the categorization of patterns, the pattern format employed, as well as the process of assigning UX factors. Sections 7.6.4 through 7.6.9 discuss issues related to writing each of the sections that constitute a pattern. These lessons might prove useful for other researchers working in this area.

7.6.1 Pattern categorization

Figure 4 (section 7.4.2) categorizes the patterns identified in this research according to activity domain (such as navigation, search, and so on). This is in line with the recommendations of Segerståhl and Jokela (2006). By promoting a standard strategy for pattern categorization, it is believed that different pattern collections will become more consistent with each other and thus become easier to use in combination. During the categorization process, some difficulties related to consistency were encountered. For example, the category "information management" includes the pattern Tag-Based Navigation, which is about information management (i.e. tagging information by supplying specific keywords), but also about navigation (i.e. using tag clouds as a way of navigating through information). There were 4 other patterns related to information management, but none related to navigation. Thus I could either invent a new category called "navigation" for this single pattern, or I could choose to put it into the already existing category "information management". If different pattern collections are to be organized in a consistent manner so as to make them easier to use in combination, it seems clear that more detailed guidelines for categorization are needed.

Van Welie & Van Der Veer (2003) suggested an alternative approach for grouping patterns, based on grouping principles borrowed from object-oriented modeling[19]. This approach is primarily suited to patterns that have a hierarchical structure. Since the patterns presented in this research mostly address issues of similar scope, they are not suitable for this type of organization.

7.6.2 Pattern format

The pattern format I have employed conforms to the one presented in Karahasanovic et al. (2008). This is a fairly simple format which in turn has been borrowed from Common Ground (Tidwell, 1999). Structuring patterns in this way was relatively straightforward, but the format was sometimes felt as being a bit constraining. The lack of a dedicated *references* section, made it necessary to put links to any related patterns in the solution section instead. Although

[19] This approach is described in more detail in section 3.2.1.

this is an acceptable workaround, pattern readability could have been slightly increased with the addition of a dedicated references section.

7.6.3 Assigning UX factors

Assigning UX factors to patterns is not a trivial task. Since the relationship between UX factors and patterns is only briefly outlined (see section 6.1), I frequently had to make approximations when classifying patterns. For example, UX3 Motivation is defined as meaning whether the user is motivated to use the system several times. However, there are other types of motivation that are also relevant for the present context, such as motivation to create and share content. Thus patterns that deal with motivation were assigned this UX factor even if they did not deal specifically with motivating repeated use of the system. If more detailed guidelines for assigning UX factors were available, this process would have been more straightforward and would have resulted in more precisely classified patterns.

7.6.4 Naming

Finding a suitable name is a challenging aspect of the pattern development process. It was a challenge to create pattern names that were both understandable and short, while at the same time maintaining evocativeness. Pattern names were often revised during the identification process. Often an initial name was selected which captured what the pattern was about, but often containing too many words. This name was then shortened as much as possible, which often required completely re-thinking the name. All patterns adhere to the Noun Phrase Name pattern by Meszaros and Doble (1996).

7.6.5 Problem

Once the purpose of the pattern was identified, the problem section was straightforward to write. It usually involved rephrasing the goal of the pattern as a question (i.e. how can users attain goal *x*?), and ensuring that the goal was relevant from a user perspective.

7.6.6 Forces

The forces section was one of the most difficult areas of writing up a pattern, because it required a thorough analysis of possible outcomes of applying the pattern. Such an analysis formed the basis of identifying potential issues that needed to be optimized or balanced in order to arrive at a satisfactory solution.

7.6.7 Context

The context section was fairly straightforward to write. It involves identifying a fitting context for the pattern, usually related to a potential user.

7.6.8 Solution

It was often difficult to find the right amount of details to include in the solution, in order to ensure that it would provide useful information without becoming too specific to one particular implementation. There are no hard answers to this issue, and very little seems to have been written about this particular aspect of pattern writing. Looking at other pattern collections such as Common Ground (Tidwell, 1999) and The Interaction Design Pattern

Library (Van Welie, 2008), it seems that a standard length for the solution varies between 50 to 200 words, which is similar to the patterns identified in this work.

7.6.9 Examples

The example section proved to be problematic: The pattern definition presented in Karahasanovic et al. (2008) states that if examples of successful use can not be found, the pattern is either not good, or is rarely applied. This premise is relevant for patterns that are mined from existing functionality, such as the patterns in Common Ground (Tidwell, 1999). For such a collection, examples of actual use of the pattern strengthen its validity, since it provides references to the sources from which it was identified. However, the patterns identified in this research are mainly grounded in user feedback related to a specific application (i.e. Talhonia or Urørt) and, to a lesser extent, theory. In this context, the supplement of examples of successful use in other applications become less interesting, because it does not serve to validate the pattern. In the present context, the use of a *rationale* section might have been more appropriate, as it would allow each pattern to include the findings (i.e. user feedback or specific functionality) that led to its identification. As a result of this difficulty, the examples used are sometimes very specific, and might not be very useful for other readers.

8 Conclusions and future work

As software systems become increasingly complex, system designers are looking for tools that can simplify the development process. To this end, design patterns might be useful for capturing design knowledge in a way which facilitates reuse. Within software engineering, design patterns have been used since the early nineties. More recently, they have been applied in the context of HCI as well. The goal of this research was to identify user experience patterns that support the creation of applications enabling non-professional users to create and share content with others.

Although several pattern collections already exist within HCI (Borchers, 2001; Duyne et al., 2006; Tidwell, 1999, 2005; Van Welie, 2008), these collections are primarily focused on usability-related issues. At the same time, there is a lack of patterns that address the emotional and social aspects of human-technology interactions that is embodied in the user experience perspective. The present research represents a first step towards addressing this situation.

8.1 Contributions

This section presents the main contributions of this research.

8.1.1 Development of guidelines for writing patterns

Although a definition of user experience patterns was given in Karahasanovic et al. (2008), this was felt to be insufficient for practical purposes. To this end, I developed a set of guidelines for writing patterns based on the aforementioned definition and a literature review. The resulting guidelines provide more detailed information on how patterns should be written and what they should contain.

8.1.2 Analysis of Citizen Media user experience patterns

Prior to identifying my own patterns, I performed an analysis of user experience patterns identified in a previous phase of the Citizen Media project (Karahasanovic et al., 2008). The problems identified in this analysis resulted in a list of recommendations for improvement, which could be useful in a subsequent revision of these patterns. The lessons learned from this analysis provided some useful insight towards identifying new user experience patterns.

8.1.3 Identification of new user experience patterns

Two studies were conducted in order to evaluate the Citizen Media application Talhonia. The first study was conducted over a period of 2 days at SINTEF in June 2008 with 12 adults as participants. The second study was conducted over a period of 9 days at Ila primary school in January 2009 with 35 eight year old children as participants. I attended the latter study on 3 days, making observations as well as helping with practical aspects of conducting the evaluation. I performed an analysis of the findings from both studies, which formed the basis for identifying new user experience patterns. I identified 12 new user experience patterns as a result of this process.

Since all user experience patterns identified within the Citizen Media are meant to be presented as a unified collection of patterns, the next phase of the pattern identification

process involved merging the current set of patterns with those that were identified previously (Karahasanovic et al., 2008). For the merging process, I selected a set of 6 patterns that had been identified at SINTEF previously, as the remaining 20 patterns were identified by another research group. During the merging process, I focused on minimizing overlap between patterns, as well as ensuring that they conformed to a common set of guidelines. To this end, the recommendations for improvement identified earlier proved useful. After the merging was conducted, I ended up with a final set of 16 user experience patterns.

8.1.4 Comparison with existing pattern collections

I performed a comparison of the user experience patterns identified in this research with 3 pattern collections used in HCI: "Common Ground" (Tidwell, 1999), "The Interaction Design Pattern Library" (Van Welie, 2008), and "The Design of Sites" (Duyne et al., 2006). These collections contained 291 design patterns in total. The results of the comparison show that 4 patterns had some degree of overlap, 3 patterns had a high degree of overlap, while no overlap was found for the remaining 9 patterns. Thus, almost half of the patterns identified in this research replicate previous findings, indicating that they might be useful in other situations as well.

8.2 Future work

The research presented in this thesis opens some avenues for further research, particularly:

- Testing whether the identified patterns are useful in practice
- Testing the validity of patterns in HCI in general

8.2.1 Testing whether the identified patterns are useful in practice

Although the findings of the pattern comparison conducted in section 7.5 indicate that the patterns identified in this work have some support elsewhere, further research is needed in order to draw any conclusions regarding the quality or usefulness of these patterns. A natural first step in this regard is to test whether the patterns are readily understood by potential users. This could be investigated by presenting the patterns to a group of people, and administering a questionnaire directly afterwards to test whether key aspects of the patterns are understood or not.

A possible next step in the evaluation of the user experience patterns is to investigate whether they bring benefits when used in practice. One possible way of evaluating this is to conduct a controlled experiment where two groups of participants receive a design task, such as designing a paper prototype of an application that fits within the vision of the Citizen Media project. The experimental group would be given the user experience patterns, while the control group would complete their task without the help of patterns. Comparing the performance of the two groups, which could be measured by an independent evaluation of the resulting prototypes, the difference between the experimental group and control group would give an indication of whether the use of these patterns lead to better results in a practical design task.

8.2.2 Testing the validity of patterns in HCI in general

Despite the amount of research activity related to design patterns in HCI in later years, one important area that has received little attention is the evaluation of design patterns' usefulness in practice (Dearden & Finlay, 2006). In order to test this, a comparison with alternative approaches could be conducted. To my knowledge, only Cowley and Wesson's (Cowley & Wesson, 2005) study has attempted this as of yet. Although a full analysis of their findings has not been conducted, their preliminary findings suggest that participants are more positive towards design patterns than guidelines. Nonetheless, more research is needed in order to draw any conclusions regarding this.

Some additional insight can be gained from the field of software engineering, where more research has been conducted on this topic (Bieman, Jain, & Yang, 2001; Bieman, Straw, Wang, Munger, & Alexander, 2003; Prechelt, Unger, Tichy, Brossler, & Votta, 2001; M. Vokáč, 2004; Marek Vokáč, Tichy, Sjøberg, Arisholm, & Aldrin, 2004). Although the use of design patterns in software engineering is different from their use in HCI, one general finding with software engineering design patterns was that the benefit of applying patterns seemed to depend on the quality of the pattern itself (including how easy it was to understand for the pattern user). For example, the studies by Vokáč (2004) and Prechelt et al. (2001) found some patterns to be associated with high success rates while others had low success rates, possibly due to the latter patterns being more difficult to understand and apply correctly. If these findings are valid for HCI as well, they indicate that if design patterns are to be useful in practice, it's wise to spend some extra time gathering feedback from potential users and revising the patterns accordingly in order to optimize understandability and clarity.

Bibliography

Alexander, C. (1977). *A Pattern Language: Towns, Buildings, Construction*: Oxford University Press, USA.

Appleton, B. (1997). Patterns and software: essential concepts and terminology. *Object Magazine Online, 3*(5), 16.

Bayle, E., Bellamy, R., Casaday, G., Erickson, T., Fincher, S., Grinter, B., Gross, B., Lehder, D., Marmolin, H., Moore, B., Potts, C., Skousen, G., & Thomas, J. (1998). Putting it all together: towards a pattern language for interaction design: A CHI 97 workshop. *SIGCHI Bull., 30*(1), 17-23.

Bennedsen, J., & Eriksen, O. (2003). *Applying and developing patterns in teaching.* Paper presented at the Frontiers in Education, 2003. FIE 2003. 33rd Annual.

Bieman, J. M., Jain, D., & Yang, H. J. (2001). *OO design patterns, design structure, and program changes: an industrial case study*.

Bieman, J. M., Straw, G., Wang, H., Munger, P. W., & Alexander, R. T. (2003). *Design patterns and change proneness: an examination of five evolving systems*.

Borchers, J. (2000). *A pattern approach to interaction design*, New York City, New York, United States.

Borchers, J. (2001). *A Pattern Approach to Interaction Design*: Wiley.

Boyd, D. M., & Ellison, N. B. (2008). Social Network Sites: Definition, History, and Scholarship. *Journal of Computer-Mediated Communication, 13*(1), 210-230.

Cowley, N. L. O., & Wesson, J. L. (2005). An Experiment to Measure the Usefulness of Patterns in the Interaction Design Process *Human-Computer Interaction - INTERACT 2005* (pp. 1142-1145).

Dearden, A., & Finlay, J. (2006). Pattern languages in HCI: A critical review. *HUMAN-COMPUTER INTERACTION, 21*(1), 49-102.

Dearden, A., Finlay, J., Allgar, L., & McManus, B. (2002). *Evaluating pattern languages in participatory design.* Paper presented at the CHI '02 extended abstracts on Human factors in computing systems, Minneapolis, Minnesota, USA.

Duyne, D. K. v., Landay, J. A., & Hong, J. I. (2006). *The Design of Sites: Patterns for Creating Winning Websites*: Prentice Hall.

Erickson, T. (2000). *Lingua Francas for design: sacred places and pattern languages*, New York City, New York, United States.

Fincher, S. (2000). HCI Pattern-Form Gallery Retrieved 29.09.2008, from http://www.cs.kent.ac.uk/people/staff/saf/patterns/gallery.html

Fincher, S. (2003). CHI 2003 Workshop Report Perspective on HCI Patterns: Concepts and tools (introducing PLML). *Interfaces, 56*, 27-28.

Fowler, M. (2006). Writing Software Patterns Retrieved 17.09.2008, from http://martinfowler.com/articles/writingPatterns.html

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. M. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*: Addison-Wesley Professional.

Hassenzahl, M. (2008). *User experience (UX): towards an experiential perspective on product quality.* Paper presented at the Proceedings of the 20th International Conference of the Association Francophone d'Interaction Homme-Machine, Metz, France.

Hassenzahl, M., & Tractinsky, N. (2006). User experience - A research agenda. *Behaviour and Information Technology, 25*(2), 91-97.

Karahasanovic, A., Wurhofer, D., Obrist, M., Tscheligi, M., Schwaiger, D., Riegler, S., & Ruud, K. A. (2008). *A/V networks user experience patterns and methods*.

Kotzé, P., Renaud, K., & Biljon, J. v. (2008). Don't do this - Pitfalls in using anti-patterns in teaching human-computer interaction principles. *Computers & Education, 50*(3), 979-1008.

Law, E. L.-C., Roto, V., Hassenzahl, M., Vermeeren, A. P. O. S., & Kort, J. (2009). *Understanding, scoping and defining user experience: a survey approach.* Paper presented at the Proceedings of the 27th international conference on Human factors in computing systems, Boston, MA, USA.

Lüders, M., Karahasanovic, A., Pelt, M., & Herreweghe, H. V. (2008). *Designing for communities: Collaborative storytelling in Talhonia*.

Mahemoff, M. J., & Johnston, L. J. (1998). *Principles for a usability-oriented pattern language*.

McGee, K. (2007). *Patterns and computer game design innovation.* Paper presented at the Proceedings of the 4th Australasian conference on Interactive entertainment, Melbourne, Australia.

Meszaros, G., & Doble, J. (1996). Metapatterns: A pattern language for pattern writing. The 3rd Pattern Languages of Programming conference. *In 3rd Pattern Languages of Programming conference*.

Miller, G. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *Psychological Review, 63*(2), 81-97.

Pemberton, L. (2000). *The Promise of Pattern Languages for Interaction Design*. Paper presented at the Human Factors Symposium. Retrieved 10.03.2009, from http://www.cmis.brighton.ac.uk/staff/lp22/HF2000.html

Prechelt, L., Unger-Lamprecht, B., Philippsen, M., & Tichy, W. F. (2002). Two controlled experiments assessing the usefulness of design pattern documentation in program maintenance. *Software Engineering, IEEE Transactions on, 28*(6), 595-606.

Prechelt, L., Unger, B., Tichy, W. F., Brossler, P., & Votta, L. G. (2001). A controlled experiment in maintenance: comparing design patterns to simpler solutions. *Software Engineering, IEEE Transactions on, 27*(12), 1134-1144.

Salingaros, N. (2000). The structure of pattern languages. *Architectural Research Quarterly, 4*, 149-161.

Seffah, A. (2003). Learning the ropes: human-centered design skills and patterns for software engineers' education. *interactions, 10*(5), 36-45.

Segerståhl, K., & Jokela, T. (2006). *Usability of interaction patterns*, Montréal, Québec, Canada.

Tidwell, J. (1999). Common Ground Retrieved 05.09.2008, from http://www.mit.edu/~jtidwell/common_ground.html

Tidwell, J. (2005). *Designing Interfaces: Patterns for Effective Interaction Design*: O'Reilly Media, Inc.

Van Welie, M. (2008). Interaction Design Pattern Library Retrieved 06.09.2008, from http://www.welie.com/patterns/index.php

Van Welie, M., & Van Der Veer, G. C. (2003). *Pattern Languages in Interaction Design: Structure and Organization*.

Van Welie, M., Van Der Veer, G. C., & Eliëns, A. (2000). Patterns as Tools for User Interface Design.

Verde, O., Karahasanovic, A., Bruynooghe, B., Herreweghe, H. V., Broeck, M. V. D., Brandtzæg, P. B., Vermeir, L., & Vanattenhoven, J. (2008). *Talhonia: empowering collaborative storytelling*.

Vlissides, J. (1996). Hillside.net - Papers - 7 Habits, from http://www.hillside.net/patterns/papers/7habits.html

Vokáč, M. (2004). Defect frequency and design patterns: An empirical study of industrial code. *IEEE Transactions on Software Engineering, 30*(12), 904-917.

Vokáč, M., Tichy, W., Sjøberg, D. I. K., Arisholm, E., & Aldrin, M. (2004). A Controlled Experiment Comparing the Maintainability of Programs Designed with and without Design Patterns—A Replication in a Real Programming Environment. *Empirical Software Engineering, 9*(3), 149-195.

Winn, T., & Calder, P. (2002). Is this a pattern? *Software, IEEE, 19*(1), 59-66.

Woods, D. D., Patterson, E. S., & Roth, E. M. (2002). Can We Ever Escape from Data Overload? A Cognitive Systems Diagnosis. *Cognition, Technology & Work, 4*(1), 22-36.

Wurhofer, D., Beck, E., Obrist, M., Kepplinger, S., Moser, C., Igelsböck, J., Förster, F., Tscheligi, M., Santangelo, A., & Lüders, M. (2008). *User Experience Evaluation*.

Appendix A – Existing patterns in Citizen Media

The patterns identified in an earlier phase of the Citizen Media project are listed below, taken from Karahasanovic et al. (2008). The patterns identified by UoS ICT&S (20 in total) are displayed first, followed by the patterns identified at SINTEF (6 in total).

Patterns developed at UoS ICT&S

1. Forming groups of interest (UX6 Co-Experience, UX7 Sociability)

Name: Main idea of the pattern in a few words + UX factor addressed	Forming groups of interest (UX6 Co-Experience, UX7 Sociability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How can the forming of groups of users with shared interests be supported through an A/V application?
Forces: Further elaboration of problem statement	- Users want to be able to meet other users with similar interests - Users want to be able to form "special interest groups" - Users want to be able to subscribe to "special interest groups" - Users want to exchange specific content with others - Users want to tell about their experiences and get to know other users experiences (information exchange) - Users want offline communities to be reflected online in an A/V application
Context: Characteristics of the context of use	Forming of interest groups is important for networked applications in order to boost Co-Experience and Sociability. Online interest groups reflect participation in offline communities.
Solution: Successful solution for the described problem	Integrate functions for forming and subscribing to special interest groups. Promote these groups within the application (give information about them). Enable suggestion or posting of topics for a certain group Enable invitation of other people
Examples: Examples of successful use of the pattern in a system	Subscribing to "special interest groups" in Flickr, Facebook Offline communities in Engerwitzdorf ask for the possibility to launch groups of interest on the IPTV platform (reflecting the communities).

2. Share Content (UX7 Sociability)

Name: Main idea of the pattern in a few words + UX factor addressed	Share Content (UX7 Sociability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How can the sharing of content be supported by an A/V application?
Forces: Further elaboration of problem statement	- Users want to belong to the community - Users want to have "collective" content which they can share with other users
Context: Characteristics of the context of use	Having some collective content which is shared among the users strengthens the community and lets users feel part of the community. Through the resulting human-human interaction, a positive experience of sociability is supported.
Solution: Successful solution for the described problem	Support the sharing of content by providing features for uploading, viewing and modifying "collective" content.
Examples: Examples of successful use of the pattern in a system	Sharing of videos on the IPTV platform of Engerwitzdorf

3. Create Content (UX1 Fun, UX3 Motivation, UX5 Engagement)

Name: Main idea of the pattern in a few words + UX factor addressed	Create Content
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How can users be encouraged to create content and place it on the platform?
Forces: Further elaboration of problem statement	- Active contribution of users let them feel as being part of the community - Users want to produce content by themselves and to show it on the platform
Context: Characteristics of the context of use	User experience is often increased when users are allowed to act as "designers" and to actively contribute to the appearance and the contents of the platform.
Solution: Successful solution for the described problem	Give users "tools" that enable content production and placement on the platform.

Examples: Examples of successful use of the pattern in a system	Creating videos for the IPTV platform of Engerwitzdorf

4. Co-create Content (UX6 Co-Experience, UX7 Sociability)

Name: Main idea of the pattern in a few words + UX factor addressed	Co-create Content (UX6 Co-Experience, UX7 Sociability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How can users be encouraged to create and modify community content together?
Forces: Further elaboration of problem statement	- Collectively creating content strengthens the community aspect - Active contribution of users let them feel as being part of the community - Users want to be actively involved in the content creation of the application
Context: Characteristics of the context of use	Creating content together with other users can be an important goal of the platform, as it provokes co-experience and sociability.
Solution: Successful solution for the described problem	Provide features for collectively creating and modifying content
Examples: Examples of successful use of the pattern in a system	Creating stories in the Talhonia application; "Unort" application: defining and voting Unorte together

5. Challenge Users (UX1 Fun, UX2 Motivation)

Name: Main idea of the pattern in a few words + UX factor addressed	Challenge users (UX1 Fun, UX2 Motivation)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to make A/V application more fun and raise the motivation to use the application?
Forces: Further elaboration of problem statement	- Users enjoy competing with other users - Users want to measure/compare their skills - Users strive for variety - Users want to interact with others via competitions
Context: Characteristics of the context of use	Challenging others is enjoyed by users and motivates them to use the application. What is more, sociability is supported.
Solution:	Add challenging elements like games, quizzes, competitions.

Successful solution for the described problem	Different difficulty levels address more users and are more fun as users can raise the difficulty level. A voting function can also be challenging.
Examples: Examples of successful use of the pattern in a system	Game, smiley competition, quiz for Engerwitzdorf community, voting function of Kabadu.

6. Make Information Management Easy (UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Make Information Management Easy (UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to handle large amounts of content?
Forces: Further elaboration of problem statement	- Users often have much content to administer - Users want to find stored information or objects easy and quickly - Users want to group content by topics - Users want to see their stored data at a glance
Context: Characteristics of the context of use	Networked applications often have a large amount of user-data to be managed, and so it is important to manage these data efficiently.
Solution: Successful solution for the described problem	Let users manage sets of objects like photos and videos by using overviews and detail-views. Integrate features for easily adding new content and deleting waste content. Give users the possibility to group objects by making different folders and sub-folders and showing them in a tree structure for a better overview. Integrate a search function for searching through the content.
Examples: Examples of successful use of the pattern in a system	CITIZEN MEDIA platform of Engerwitzdorf – users have to manage their “friends”, videos, photos, … They have an overview of these objects and can change to the detail-view whenever they want.

7. Make It Enjoyable (UX1 Fun)

Name: Main idea of the pattern in a few words + UX factor addressed	Make It Enjoyable (UX1 Fun)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to let users have fun or feel entertained when using A/V networks?
Forces: Further elaboration of problem statement	- Users are looking for entertainment when using an A/V networks - Users want to enjoy using the A/V application
Context: Characteristics of the context of use	Promoting the experience of fun lets the user associate the networked application with positive feelings and emotions and therefore supports frequent use.
Solution: Successful solution for the described problem	Add challenging and surprising elements to your site, supported by additional visual fun-adding details, to create a highly interactive and visual experience. Add enjoyable elements – for example, enable sending gifts to other users. Add entertaining features.
Examples: Examples of successful use of the pattern in a system	Games, topics concerned with fun, quiz

8. Finding Information (UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Finding Information (UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to help users to find information (related to a specific topic) quickly and efficiently?
Forces: Further elaboration of problem statement	- Users strive for information which is easy and fast to find - Users do not want to invest much time in searching for specific contents
Context: Characteristics of the context of use	Networked applications often contain large amounts of data and information which should not be too time-consuming or difficult to find.
Solution: Successful solution for the described problem	Primarily allow users to browse the information but combine it with more specific search tools that support other types of searching. Tagging / tag clouds. Links between objects (recommendations).
Examples: Examples of successful use of the pattern in a system	Search of cabaret artists or cabaret performances on the Kabadada platform.

9. Getting to know the application (UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Getting to know the application (UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to let users learn about the right handling of the networked application at the beginning of their use?
Forces: Further elaboration of problem statement	- Novice users want to learn how to use the platform quickly - Users prefer to learn interactively - Learning how to handle the platform at the beginning is important
Context: Characteristics of the context of use	Actively support the right handling of the platform.
Solution: Successful solution for the described problem	Construct a learning experience from the basic learning tasks that is in line with your audience and site. Life events, (interactive) tutorials, workshops. Manual
Examples: Examples of successful use of the pattern in a system	Users learn how to handle the CITIZEN MEDIA IPTV platform by integrating the learning process in games or by giving them videos about how to use the platform.

Provide feedback (UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Provide feedback (UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to inform users about the status and results of their operations and actions?
Forces: Further elaboration of problem statement	- Users have to get informed about errors in order to learn from them. - Users want to know whether or not the operation is still being performed as well as how much longer the user will need to wait. - Users want to get reactions of the system as responses to their actions - A user wants to be sure that their action has a result, otherwise the user is likely to think that the system has crashed or that he has acted incorrectly
Context: Characteristics of the context of use	Getting no feedback to performed actions from the system lets users get confused, ambiguous or switch to other actions.

Solution: Successful solution for the described problem	Show error messages, warnings and processing advancement. Show final status and/or results. Show that the application is still working and give an indication of the progress. Provide feedback at a rate that gives the user the impression that the operation is still being performed e.g. every 2 seconds using animation. Additionally, provide a valid indication of the progress. Progress is typically the remaining time for completing, the number of units processed or the percentage of work done. For example, the progress can be shown by using a progress bar or a sandglass. Create consistent and predictable feedback , meaning that all elements react in the same manner.
Examples: Examples of successful use of the pattern in a system	Status information from Engerwitzdorf IPTV platform.

10. Provide Personal Information (UX5 Involvement)

Name: Main idea of the pattern in a few words + UX factor addressed	Provide Personal Information (UX5 Involvement)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to support non-anonymous interaction between users?
Forces: Further elaboration of problem statement	- Playing against friends is more enjoyable than playing against strangers - Users want to know with whom they are interacting - Users want to identify their interaction partner - Users want to get to know other users - Users want to present themselves to others
Context: Characteristics of the context of use	Interacting with users whose characteristics are well-known is more interesting than interacting with anonymous users and therefore lets involvement increase.
Solution: Successful solution for the described problem	Let users create and view personal profiles. Let users choose friends (friend list). Give information about the status of friends. Provide the possibility to upload personal photos, videos or text on the platform.
Examples: Examples of successful use of the pattern in a system	The CITIZEN MEDIA IPTV/web-application allows to create and show a personal profile and to have a friend list to which friends can be added.

12. Successful navigation (UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Successful navigation (UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to make navigation on the platform easy and quick?
Forces: Further elaboration of problem statement	- Users want to reach certain goals easily. - Users want to have an orientation on the platform (where they are and how they come back)
Context: Characteristics of the context of use	Navigation on the platform is a major component of the overall usability of the application.
Solution: Successful solution for the described problem	Let the user always know where he is,where he comes from and how he can return; Good and approved navigation structure. Back-arrow, Home button, map, breadcrumbs.
Examples: Examples of successful use of the pattern in a system	Negative-example: The navigation on the IPTV platform in Engerwitzdorf is difficult for users and can cause serious problems.

13. Stimulate social interaction (UX7 Sociability, UX6 Co-Experience)

Name: Main idea of the pattern in a few words + UX factor addressed	Stimulate social interaction (UX7 Sociability, UX6 Co-Experience)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to enhance the communication between users?
Forces: Further elaboration of problem statement	- Users want to interact socially with other users via the platform - Users want to tell other users about their life
Context: Characteristics of the context of use	The possibility of interacting socially with other users represents a major reason for using an A/V networked application.

Solution: Successful solution for the described problem	Offer different means of communication (chat, messages, blogs, …). Games and team play. Visualization of social interactions. Rules/protocols for communication.
Examples: Examples of successful use of the pattern in a system	Photo- and videosharing in Engerwitzdorf. Kabadada Blog. Internal messages (Kabadada).

14. Share an Experience (UX6 Co-Experience, UX7 Sociability)

Name: Main idea of the pattern in a few words + UX factor addressed	Share an Experience (UX6 Co-Experience, UX7 Sociability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to enable the sharing of experiences with other users?
Forces: Further elaboration of problem statement	- Users want to get in contact with other users - Users want to let other users partake or profit from their experiences - Users want to get information by knowing about experiences of others
Context: Characteristics of the context of use	Sharing of experiences enhances co-experience and sociability.
Solution: Successful solution for the described problem	Provide mechanisms for exchanging experiences: discussion boards, photo or video upload mechanisms. Blogging, chat, text messages, messaging services; Means for communicating mood: mood messages, emoticons, music, colors
Examples: Examples of successful use of the pattern in a system	Photo – and video sharing of the CITIZEN MEDIA platform

15. Feeling as part of the community (UX7 Sociability)

Name: Main idea of the pattern in a few words + UX factor addressed	Feeling as part of the community (UX7 Sociability)
Problem: Problems related to the usage of the system which are solved/improved	Users often do not have the feeling of "belonging to a community"

by the pattern	
Forces: Further elaboration of problem statement	- Support forming of a sense of community - Stimulate social interaction/communication between users
Context: Characteristics of the context of use	Feeling as part of a certain community increases sociability.
Solution: Successful solution for the described problem	Support forming of "social groups" by gaming competitions (games) which require team play and cooperation (sharing of resources, coordinate actions). Support forming of teams and common goals. Give users common tasks and goals. Let users create an own language and own elements/characteristics.
Examples: Examples of successful use of the pattern in a system	CITIZEN MEDIA trophy (for project partners); life events with games

16. Catch the user (UX3 Motivation)

Name: Main idea of the pattern in a few words + UX factor adressed	Catch the user (UX3 Motivation)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to motivate the use of a networked application for the first time?
Forces: Further elaboration of problem statement	- Users must be attracted by the platform so that they spend time on using it - Users want to know why they should spend time on the platform - Using the platform should be valuable for the users.
Context: Characteristics of the context of use	A networked platform has to be introduced and promoted (in an appropriate way) so that users are interested in it and will use it for a first time.
Solution: Successful solution for the described problem	Present the platform with its main characteristics to potential users. Define a clear aim/goal/purpose of the A/V networked system (identification of the user with the platform let motivation increase). Games, information, interest groups, polls… Promote the platform. Attractive design. Add gratifications and value for using the platform.
Examples: Examples of successful use of the pattern in a system	The voting function of Kabadu (represents an incentive for many amateur cabaret artists).

17. Keep the user active (UX3 Motivation)

Name: Main idea of the pattern in a few words + UX factor adressed	Keep the user active (UX3 Motivation)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to motivate the user to continue using a networked application?
Forces: Further elaboration of problem statement	- Users need incentives for using the platform - Users want variety when using the platform
Context: Characteristics of the context of use	Users who are bored by the platform will not use it any more, and therefore it is important to avoid boredom when using the application.
Solution: Successful solution for the described problem	Continuously add new content and features Stay up to date, Further development of application Let users actively participate (the platform) Provide a broad spectrum of features
Examples: Examples of successful use of the pattern in a system	Newest video on start page of Kabadada; News section of Kabadada which is prominently placed on the starting page; Smiley completion (collecting smileys) in Engerwitzdorf

18. Encourage user generated content (UX3 Motivation)

Name: Main idea of the pattern in a few words + UX factor adressed	Encourage user generated content (UX3 Motivation)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to motivate users to generate content ?
Forces: Further elaboration of problem statement	- Users need motivation for generating and implement own ideas.
Context: Characteristics of the context of use	User participation in the form of content creation should be increased.
Solution: Successful solution for the described problem	Reward users for generating content themselves Presentation of generated content Place new content prominently Rating of user generated content
Examples: Examples of successful use of the	Smiley tasks of CITIZEN MEDIA platform – for generating

pattern in a system	content, users are rewarded with smileys; Kabadada & Kabadu: the newest video is placed on the start page of the platform

19. Vote to promote (UX7 Sociability)

Name: Main idea of the pattern in a few words + UX factor adressed	Vote to promote (UX7 Sociability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	The user wants to promote a particular piece of content in a community pool of submissions.
Forces: Further elaboration of problem statement	- The promotion of a particular piece of content takes the form of a vote for that item, and items with more votes rise in the rankings to be displayed with more prominence.
Context: Characteristics of the context of use	Use this solution when … … users in the community have the ability to submit content to a 'pool' of resources … some democratic form of judgment is needed, to allow the community to compare the subjective quality of one submission to another … a sizeable-enough community is required. Ideally, popular submissions in the pool should receive significantly more (dozens, hundreds?) votes than on-popular ones, in order to make comparisons meaningful.
Solution: Successful solution for the described problem	Provide a voting mechanism; Highlight popular items; Prominently display the number of votes that an item has received (see link for more info!!!)
Examples: Examples of successful use of the pattern in a system	Voting of videos on the Kabadu platform

20. Give information about the platform (UX3 Motivation, UA4 Trust)

Name: Main idea of the pattern in a few words + UX factor addressed	Give information about the platform (UX3 Motivation, UA4 Trust)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	Users want to be informed about the whole purpose of the platform, especially when they are novel users.
Forces: Further elaboration of problem statement	- Users want to have background information concerning the application - Users want to know the target audience of the application immediately - Users want to know the intention / purpose of the application
Context: Characteristics of the context of use	Enough information about the platform increases motivation and trust and therefore also the probability to use it.
Solution: Successful solution for the described problem	Give information about the platform and its main characteristics, so that people immediately know if the platform meets their expectations and requests.
Examples: Examples of successful use of the pattern in a system	Negative-example: Test-users of the Kabadada platform did not get information about the main characteristics of this platform and were confused and angry about that fact.

Patterns developed at SINTEF

21. Getting in touch with people sharing the same interests (UX5 User involvement, UX6 Co-Experience, UX7 Sociability)

Name: Main idea of the pattern in a few words + UX factor addressed	Getting in touch with people sharing the same interests (UX5 User involvement, UX6 Co-Experience, UX7 Sociability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How can getting in touch with people sharing the same interests be supported through a networked A/V application?
Forces: Further elaboration of problem statement	- Users want to comment the music that other artists uploaded, but they don't always want that their comments are publicly available - User wants to receive the feedback from other artists - Users want to contact other artists to get listeners and spark interest - Users want to initiate collaboration with other artists
Context: Characteristics of the context of use	Getting in touch with people sharing the same interests is one of the prerequisites for UX5, UX6 and UX7
Solution: Successful solution for the described problem	A networked A/V application should include as its integrated parts both publicly available massaging systems (e.g. guestbooks and walls) and private communications channels. This allows simple human-human interaction and providing feedback on others work.
Examples: Examples of successful use of the pattern in a system	Getting in touch with people sharing the same interests in music through Urørt application, for example contacting potential new band members, organising joint-concerts and play sessions

22. Motivating people to make their own music available to others (UX3 Motivation, UX5 User Involvement, UX6 Co-experience)

Name: Main idea of the pattern in a few words + UX factor addressed	Motivating people to make their own music available to others (UX3 Motivation, UX5 User Involvement, UX6 Co-experience)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How can sharing of user generated music be supported through networked A/V applications?
Forces: Further elaboration of problem statement	- Users want to acquire listeners of their own music - Users want to receive constructive feedback on their music - Users want to provide feedback to other artists - Users want to find new music; they are often interested in music from their home-town or region.
Context:	Sharing content (music) with other people and receiving

Characteristics of the context of use	feedback motivates people to create more and better music (UX3), increases their involvement (UX5) and provides a basis for co-experience (UX6).
Solution: Successful solution for the described problem	Provide a site where users can: - easily upload and promote their music - provide/receive feedback, reviews and ratings - find new and interesting music. Region based search should be provided in addition to other search types (genre, events)
Examples: Examples of successful use of the pattern in a system	Artists uploading their own music and receiving feedback from their listeners through Urørt application

23. Providing comprehensive information about a networked A/V application on its front page (UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Providing comprehensive information about a networked A/V application on its front page(UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How can a first page of an A/V application best communicate its purpose and usage to potential/existing users?
Forces: Further elaboration of problem statement	- Users want comprehensive information about a site without overwhelming or irrelevant content
Context: Characteristics of the context of use	Comprehensive information about an A/V application improves its usability (UX8).
Solution: Successful solution for the described problem	The front page should be built with well structured categorizing entities, such as tags and genres. Concepts used on navigation buttons, links and symbols should be carefully designed. The front page should clearly communicate the concept and content of the site, and log-in area of the community.
Examples: Examples of successful use of the pattern in a system	Information about Urørt application

24. Navigation within a networked A/V application (UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Navigation within a networked A/V application (UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How to find relevant content within a networked A/V application?
Forces: Further elaboration of problem statement	- Users want an easy way to search and navigate through the available content; navigation is particularly important for audio content as users needs more time to evaluate interestingness of the content.
Context: Characteristics of the context of use	Good navigation and search mechanisms improves usability of an A/V application (UX8).
Solution: Successful solution for the described problem	Applications should provide means for supporting navigation such us tags and tag clouds describing audio content ; As some users are not familiar with tags, a "what are tags"-link describing this concept should be provided.
Examples: Examples of successful use of the pattern in a system	Navigation mechanisms of Urørt application

25. Personalisation of user profiles (UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Personalisation of user profiles (UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How personalization of user profiles should be supported by A/V applications?
Forces: Further elaboration of problem statement	- Users want to personalize the look of their profiles; it is particularly important for users to be able to communicate the genre and type of music they play - Users wants consistent solutions for editing their profiles, use of Norwegian common word instead of technical concepts in English (for example widget), and automatic confirmation when editing.
Context: Characteristics of the context of use	Personalisation of user profiles increases usability of A/V applications (UX8).
Solution: Successful solution for the described problem	The application should provide simple functionality for editing user profiles allowing users to make their personal design (choose background, patterns, colours, fonts, including photos and web widgets). A compromise should be made between user-freedom and guiding constraints. Norwegian common words should be used instead of technical concepts such as

	widget. Automatic confirmation should be given when a change done (“Your photo has been uploaded”).
Examples: Examples of successful use of the pattern in a system	Possibilities to personalize their user profiles at Urørt application

26. Keeping users updated (UX5 User involvement, UX3 Motivation, UX8 Usability)

Name: Main idea of the pattern in a few words + UX factor addressed	Keeping users updated (UX5 User involvement, UX3 Motivation, UX8 Usability)
Problem: Problems related to the usage of the system which are solved/improved by the pattern	How should automatic updates be supported by A/V applications?
Forces: Further elaboration of problem statement	- Users want to receive/provide comments on newly updated content - Users want automatic news updates from friends - Users want automatic updates on their own activities - Users want more intuitive and simple system than subscription to RSS-feeds
Context: Characteristics of the context of use	Functionality for keeping users updated increases the usability of A/V system (UX8), increases users’ involvement (UX5) and motivates users to contribute with their content (UX3).
Solution: Successful solution for the described problem	The application should provide an easy way of sharing information about updates and new content. A simple functionality for subscribing on automatic news-feeds from friends and fans and an automatic notification service about new messages should be provided.
Examples: Examples of successful use of the pattern in a system	Users of the Urørt application refer to other social sites like Facebook for providing an example.

Appendix B – Summary of findings from Talhonia June study

This section lists the data gathered while analyzing the evaluator and observer-documentation, as well as the sound recordings, from the first Talhonia study, which was conducted at SINTEF on the 11th and 12th of June 2008. This section is written in Norwegian.

Logg inn og åpne en historie

- Ingen informasjon på førstesiden.
- Skjønner ikke poenget med trepanel-bakgrunnsbildet på forsiden – det gir andre forventninger enn det som oppfylles.
- Navigeringen blir uoversiktlig ved mange historier.
- Brukerveiledning mangler.
- Misnøye med at innholdet (bilder, tekst, historieliste, osv.) er fordelt på flere vinduer som flyter omkring.
- Savner vinduer som kan ”dockes”.
- Størrelsen på vinduene er ikke tilpasset innholdet. Dette gjelder for både vinduet med historielisten og selve historien.
- Manglende filtreringsfunksjoner i historielisten.
- Man kan ikke ha flere historier åpne samtidig.
- Ingen preview funksjon (editoren er ikke WYSIWYG).
- Bør ha automatisk oppdatering av nye historier i historielisten.
- Burde være lettere å se hvem som har tilgang til den aktuelle historien.
- Storylist burde vise dato for siste endring av historien samt hvem som har endret.

Redigere og lagre historie

- Markøren hopper ut av vinduet når man vil skrive.
- Lagring er problematisk.
- Save og cancel bør være tydeligere.

Utforske tidligere versjoner av historien

- Endringer burde markeres – for eksempel vha farger eller annen utheving.
- På samme måte bør det synes hvem som har skrevet hva. Forskjellige forfattere kan ha ulike bakgrunnsfarger – det kan organiseres i blokker med tilhørende datoer. Evt. kan mer informasjon vises når musepeker holdes over.
- Hvilke endringer som er gjort bør vises tydelig.
- Savner mulighet for å gå tilbake til tidligere versjon (gjør denne versjonen gjeldende).
- ”Kan man gå tilbake å kopiere og klippe inn igjen ting som var med i historien tidligere?”: Det kan gjøres vha manuell klipp og lim, men nåværende funksjonalitet ”legger opp til at man kan se hva som har skjedd, men ikke at man kan bruke det til noen faktisk nyttig funksjon”.

Kommentere en historie

- Man må bla nedover for å finne kommenteringsboksen.
- Uklart hva poenget med kommentarer er. Chat kunne kanskje funke bedre, ettersom

fokuset er på live samarbeid.
- Før man kan legge inn kommentar må man slette instruksjonsteksten, den burde forsvinne av seg selv.
- Add comment bryter med forventning: Man tror den legger til det man har skrevet, men i stedet fungerer det som en ”start new comment” knapp.
- Kommentarer burde vises direkte i teksten.

Legge inn bilde

- Det må være mulig å laste inn egne bilder.
- Bilder kommer opp i eget vindu, de burde heller vises sammen med historien de hører til. Forholdet mellom tekst og bilde blir vanskelig å se.
- Preview mangler.
- Savner integrert søkefunksjon for bilder.
- Ikke mulig å endre størrelse på bilder.
- Mangler integrasjon mot andre bildetjenester som Flickr og Facebook.

Opprette en ny historie

- Vanskelig å legge inn medforfatter.
- Administrasjon av rettigheter til medforfattere ble opplevd som vanskelig.
- Ikke mulig å jobbe på flere historier samtidig.
- Burde ikke trenge å klikke i historievinduet for at man skal kunne begynne å skrive.
- Editor vinduet kommer ikke til toppen når man klikker på det.
- Man må kunne redigere tittel.

Appendix C – Summary of findings from Talhonia January study

My observations are listed first, followed by a summary of the observations made by the other two researchers participating in the study. This summary is based on a summary document written by the main researcher conducting the study. My own observations are written in Norwegian, and the rest of the observations are a mixture of Norwegian and English.

Egne observasjoner fra evaluering av Talhonia på Ila skole, 12, 14, og 15. Januar 2009.

- Barna likte å følge med på hva de andre skrev, men de synes det var dumt at de ikke kunne skrive i samme setning som den andre holdt på med.
- Når det var mange som redigerte samtidig ble det vanskelig å finne et sted å sette inn tekst uten at man fikk en feilmelding. Barna trengte som regel hjelp når feilmeldingen dukket opp.
- Det at barna kunne se det de andre skrev førte i flere tilfeller til at de ble inspirert av hverandres ideer, om enn på en litt kronglete måte.
- Det ble litt forvirring ved innsetting av tekst på slutten av historien fordi det så ut som det ikke gikk an å legge til tekst helt på bunn. Etter hvert så jeg at det øvre vinduet måtte ”rulles” ned for å vise slutten av historien. Dette var forvirrende, og vinduet burde automatisk vise den aktuelle delen av teksten, slik at man ser sammenhengen mellom det siste som ble skrevet og det man selv vil legge til av tekst.
- Underveis i skrivingen lurte flere av barna på om det ikke ville bli mye rot når begge skrev på historien samtidig, altså at sammenhengen i historien ville forsvinne.
- To av barna mente også det var vanskelig at man så den andre skrive samtidig, for ”da brukte den andre opp alle de gode ideene”. Det kan være distraherende for barna å forholde seg til det den andre skriver og det de selv skal skrive samtidig.
- Bamsene som barna fikk utdelt virket populære, og ved flere anledninger så det ut til at barna så bort på bamsen sin når de tenkte ut hva de skulle skrive.

Sammendrag av observasjoner fra de to andre forskerne som gjennomførte studien.

- Elevene blir i liten grad frustrerte selv om Talhonia kan være litt klønete og vanskelig å bruke.
- Fortellingskortene bidrar til at elevene kommer i gang, og til at de har en fastere fortellingsramme å forholde seg til.
- Elevene løp litt mellom klasserommene for å samordne hvordan fortellingen skulle være. Sannsynligvis kan dette bli litt enklere etter hvert som det er mer tekst å arbeide med. Det ville definitivt vært en fordel med voice-chat implementert.
- Det er litt klønete å gjøre endringer i teksten. Elevene trenger hjelp underveis, og prosjektet hadde vært vanskelig å gjennomføre dersom elevene satt hjemme hos seg selv.
- Both teams worked well with continuing the stories, editing parts from yesterday, filling in details and making the stories longer. They would even have liked to write for a bit longer, and perhaps should be allowed on days with less technical troubles.
- Partcipants told they usually liked writing, but interestingly emphasised that they thought co-writing was more fun than solo-writing.

- Mats was very annoyed with the editing-window. “I have never seen a program where you can’t write in the text”. Whereas the other pupils claimed they did not find the editing-window very difficult to use, they agreed that they would rather want to write immediately in the text. Mats was also very frustrated with the difficulties of marking the right section to edit.
- The participants had the same usability-difficulties. Talhonia does offer a cumbersome way of editing text, which often results in lacking space between words. Marking the right section can also be difficult.
- Challenges with regard to co-writing were apparent. Emma wanted to write the same sentence as Erlend, and they found it difficult but fun to write at the same time. The pupils also spent some time finding the right keys.
- As with all of the other pupils who have participated, pupils found it a bit difficult to find, or understand the scroll-bar when the editing-field was visible. There are hence a number of problems with the editing-field: the pupils easily loose track of where they are editing the text, the might easier loose the context of the full story, and they find it a weird and uncommon way of editing.
- Pernille argued that the text-editing field made her loose the context of the story. Therefore she had to click insert text relatively often to see how the story made sense. She would much rather want to be able to write into the actual text. The other participants agreed.
- The teddybears play an important role for the pupils, they comfort and are cuddled with during the sessions, and clearly added an extra twist to the storywriting task.
- Interestingly, Karl used the storywriting-field to send a message to co-writing partner Viktoria, asking her to come over (a message he deleted afterwards, “because it was not part of the story”). This meeting was not to talk about the story, but confirms the impression from previous days that a chat-function should be implemented.
- Both teams were very intrigued by being able to see what the other participant was writing, confirming in the interview that this indeed was exciting.
- It was difficult for the pupils to understand why the text of the other person was marked with orange (and their own with green). I tried explaining, but in the interview, they still argued that they did not quite understand this.
- They also easily loose the context of where they are writing, and need help to move sentences around when they do not fit. This is a cumbersome process. 1) The right sentence must be marked. 2) Sentence then appears in text-editing field and can be cut out. User must then click “insert text” (while there is nothing in the editing-window). 3) User must click where the sentence should be moved. 4). Cut-out sentence is pasted into editing-field and user clicks “insert text”. This process is not very intuitive, and given the flash-interface, user must know key-commands for cut and paste.
- It is not evident how to co-write, and they often want to look at what their friend is writing before continuing. Also, they often want to write the same sentence at the same time.
- Viktoria tried to click on the back-button in the browser when she wanted to look at the story.

- Although there were no major problems with three children co-writing at the same time, more participants do make it more difficult to find somewhere in the text where they can edit. The error-message appears more frequently.